



日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2001年 5月22日

出 願 番 号

Application Number:

特願2001-152589

出 願 人

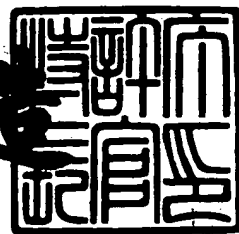
Applicant(s):

三洋電機株式会社

2001年 6月18日

特許庁長官  
Commissioner,  
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3056835

【書類名】 特許願

【整理番号】 NBC1012066

【提出日】 平成13年 5月22日

【あて先】 特許庁長官殿

【国際特許分類】 H04J 13/02

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社  
社内

【氏名】 ▲浜▼本 克昭

【特許出願人】

【識別番号】 000001889

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号

【氏名又は名称】 三洋電機株式会社

【代理人】

【識別番号】 100064746

【弁理士】

【氏名又は名称】 深見 久郎

【選任した代理人】

【識別番号】 100085132

【弁理士】

【氏名又は名称】 森田 俊雄

【選任した代理人】

【識別番号】 100091409

【弁理士】

【氏名又は名称】 伊藤 英彦

【選任した代理人】

【識別番号】 100096781

【弁理士】

【氏名又は名称】 堀井 豊

【先の出願に基づく優先権主張】

【出願番号】 特願2000-203416

【出願日】 平成12年 7月 5日

【手数料の表示】

【予納台帳番号】 008693

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0006995

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 直交符号生成装置、スクランブル符号生成装置、およびそれらを用いた携帯無線端末

【特許請求の範囲】

【請求項 1】 データの拡散処理に用いる直交符号を生成する直交符号生成装置であって、

所定の直交符号発生規則に基づいて定義される複数の直交符号系列のうち、発生すべき直交符号系列を指定する  $n$  ( $n$  は正の整数) ビットの系列番号と、前記指定された直交符号系列の順次発生すべき直交符号の位置を指定する  $n$  ビットの位置番号とを供給する手段と、

前記系列番号の最下位ビットと前記位置番号の最下位ビットとを順次論理演算することにより要素符号を発生する第 1 の論理演算手段とを備え、

前記最下位ビット同士の論理演算により発生し得る要素符号の組合せと前記組合せを反転させたものとを、前記要素符号の組合せのパターンに基づいて繰返し配列することにより前記複数の直交符号系列全体が定義され、

前記系列番号の最下位ビットを除く  $(n - 1)$  ビットと前記位置番号の最下位ビットを除く  $(n - 1)$  ビットとを順次論理演算する第 2 の論理演算手段と、

前記第 2 の論理演算手段の演算結果に対応して、前記第 1 の論理演算手段によって発生した前記要素符号または発生した前記要素符号を反転させた符号のいずれかを、前記組合せの配列に基づいて選択して前記直交符号として出力する選択手段とをさらに備えた、直交符号生成装置。

【請求項 2】 前記第 1 の論理演算手段は、前記系列番号の最下位ビットと前記位置番号の最下位ビットとの論理積を順次演算して前記要素符号として発生する論理積回路を含み、前記発生し得る前記要素符号の組合せは 3 個の符号「0」と 1 個の符号「1」とを含み、前記組合せを反転させたものは 3 個の符号「1」と 1 個の符号「0」とを含む、請求項 1 に記載の直交符号生成装置。

【請求項 3】 前記第 2 の論理演算手段は、前記系列番号の最下位ビットを除く  $(n - 1)$  ビットと前記位置番号の最下位ビットを除く  $(n - 1)$  ビットとの間で、対応する桁の値がともに 1 である桁の個数が 0 または偶数であるか、あ

るいは奇数であるかを判定する判定手段を含み、

前記選択手段は、前記判定手段による判定結果が0または偶数の場合に前記論理積回路によって発生した要素符号を直交符号として選択し、判定結果が奇数の場合に前記要素符号を反転させた符号を直交符号として選択する、請求項2に記載の直交符号生成装置。

【請求項4】 前記判定手段は、

前記系列番号の最下位ビットを除く $(n-1)$ ビットと前記位置番号の最下位ビットを除く $(n-1)$ ビットとの間で、対応する桁同士の論理積を演算する $(n-1)$ 個の論理積回路と、

前記 $(n-1)$ 個の論理積回路の演算結果を加算して、その結果が0または偶数であるか、あるいは奇数であるかを示す制御信号を発生する加算回路とを含む、請求項3に記載の直交符号生成装置。

【請求項5】 前記供給される $n$ ビットの位置番号のうち所定のビットの配列を変更させる位置番号変更手段をさらに備える、請求項1から4のいずれかに記載の直交符号生成装置。

【請求項6】 前記要素符号の組合せのパターンに基づく配列の繰返しは周期インデックス $k$  ( $k$ は正の整数)で規定され、前記複数の直交符号系列は $2^k$ 個で表わされ、前記複数の直交符号系列の各々のビット数は $2^k$ 個で表わされ、

前記位置番号変更手段は、前記 $n$ ビットの位置番号のうち下位の $k$ ビットの配列を、最下位ビットから $k$ ビット目までを逆転させるように変更する、請求項5に記載の直交符号生成装置。

【請求項7】 デジタル無線通信の携帯無線端末であって、

送信および受信のためのデータを処理するモデムと、

前記モデムの送信データを無線処理して送信無線信号として送出し、かつ受信した受信無線信号を無線処理して受信データとして前記モデムに与える無線処理装置とを備え、

前記モデムは、前記データの拡散処理に用いる直交符号を生成する直交符号生成装置を含み、

前記直交符号生成装置は、

所定の直交符号発生規則に基づいて定義される複数の直交符号系列のうち、発生すべき直交符号系列を指定する  $n$  ( $n$  は正の整数) ビットの系列番号と、前記指定された直交符号系列の順次発生すべき直交符号の位置を指定する  $n$  ビットの位置番号とを供給する手段と、

前記系列番号の最下位ビットと前記位置番号の最下位ビットとを順次論理演算することにより要素符号を発生する第 1 の論理演算手段とを含み、

前記最下位ビット同士の論理演算により発生し得る要素符号の組合せと前記組合せを反転させたものとを、前記要素符号の組合せのパターンに基づいて繰返し配列することにより前記複数の直交符号系列全体が定義され、

前記系列番号の最下位ビットを除く  $(n-1)$  ビットと前記位置番号の最下位ビットを除く  $(n-1)$  ビットとを順次論理演算する第 2 の論理演算手段と、

前記第 2 の論理演算手段の演算結果に対応して、前記第 1 の論理演算手段によって発生した前記要素符号または発生した前記要素符号を反転させた符号のいずれかを、前記組合せの配列に基づいて選択して前記直交符号として出力する選択手段とをさらに含む、携帯無線端末。

【請求項 8】 前記第 1 の論理演算手段は、前記系列番号の最下位ビットと前記位置番号の最下位ビットとの論理積を順次演算して前記要素符号として発生する論理積回路を含み、前記発生し得る前記要素符号の組合せは 3 個の符号「0」と 1 個の符号「1」とを含み、前記組合せを反転させたものは 3 個の符号「1」と 1 個の符号「0」とを含む、請求項 7 に記載の携帯無線端末。

【請求項 9】 前記第 2 の論理演算手段は、前記系列番号の最下位ビットを除く  $(n-1)$  ビットと前記位置番号の最下位ビットを除く  $(n-1)$  ビットとの間で、対応する桁の値がともに 1 である桁の個数が 0 または偶数であるか、あるいは奇数であるかを判定する判定手段を含み、

前記選択手段は、前記判定手段による判定結果が 0 または偶数の場合に前記論理積回路によって発生した要素符号を直交符号として選択し、判定結果が奇数の場合に前記要素符号を反転させた符号を直交符号として選択する、請求項 8 に記載の携帯無線端末。

【請求項 10】 前記判定手段は、

前記系列番号の最下位ビットを除く  $(n-1)$  ビットと前記位置番号の最下位ビットを除く  $(n-1)$  ビットとの間で、対応する桁同士の論理積を演算する  $(n-1)$  個の論理積回路と、

前記  $(n-1)$  個の論理積回路の演算結果を加算して、その結果が 0 または偶数であるか、あるいは奇数であるかを示す制御信号を発生する加算回路とを含む、請求項 9 に記載の携帯無線端末。

【請求項 11】 前記直交符号生成装置は、

前記供給される  $n$  ビットの位置番号のうち所定のビットの配列を変更させる位置番号変更手段をさらに備える、請求項 7 から 10 のいずれかに記載の携帯無線端末。

【請求項 12】 前記要素符号の組合せのパターンに基づく配列の繰返しは周期インデックス  $k$  ( $k$  は正の整数) で規定され、前記複数の直交符号系列は  $2^k$  個で表わされ、前記複数の直交符号系列の各々のビット数は  $2^k$  個で表わされ、

前記位置番号変更手段は、前記  $n$  ビットの位置番号のうち下位の  $k$  ビットの配列を、最下位ビットから  $k$  ビット目までを逆転させるように変更する、請求項 11 に記載の携帯無線端末。

【請求項 13】 送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置であって、

所定の生成多項式によるスクランブル符号の系列を生成するために帰還処理および拡散処理を実行するように接続された複数段のレジスタからなるシフトレジスタと、

所定の初期値をもとに前記シフトレジスタを、増加する所定回数だけそれぞれシフト動作させたならば得られる、前記帰還処理および前記拡散処理にそれぞれ関与する前記レジスタの値を演算する演算手段と、

前記演算されたレジスタの値を、対応する前記レジスタに入力する入力手段と

前記演算されかつ入力された値に基づいて前記複数段のレジスタのすべてが有効な値を保持するまで、前記演算手段が前記レジスタの値の演算を行ないかつ前記入力手段が演算された値をレジスタに入力するように前記演算手段および前記

入力手段を制御する制御手段とを備え、

前記シフトレジスタは、前記複数段のレジスタのすべてに保持された有効な値をもとにシフト動作を続行して前記スクランブル符号の系列を生成する、スクランブル符号生成装置。

【請求項 1 4】 前記所定の初期値を保持した記憶手段と、

前記所定の生成多項式に基づき前記増加する所定回数だけそれぞれシフト動作させた後における前記帰還処理および前記拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を供給する行列式供給手段とをさらに備え、

前記演算手段は、前記記憶手段に保持されている前記所定の初期値と前記行列式供給手段から供給される前記行列式とを乗算することにより前記レジスタの値を演算する、請求項 1 3 に記載のスクランブル符号生成装置。

【請求項 1 5】 前記所定の初期値を保持した記憶手段をさらに備え、

前記演算手段は、前記所定の生成多項式に基づき前記増加する所定回数だけそれぞれシフト動作させた後における前記帰還処理および前記拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を所定の演算により求め、前記記憶手段に保持されている前記所定の初期値と乗算することにより前記レジスタの値を演算する、請求項 1 3 に記載のスクランブル符号生成装置。

【請求項 1 6】 送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置であって、

所定の初期値を保持した記憶手段と、

所定の生成多項式に基づき前記スクランブル符号の系列を構成する各符号の値を決定するための行列式を所定の演算により求める論理手段と、

前記記憶手段に保持されている前記所定の初期値と、前記求められた行列式とを乗算することにより前記スクランブル符号の系列を構成する各符号の値を演算する演算手段とを備えた、スクランブル符号生成装置。

【請求項 1 7】 デジタル無線通信の携帯無線端末であって、

送信データを変調する送信系モデムと、

前記送信系モデムの送信データを無線処理して送信無線信号として送出する無線処理装置とを備え、



前記送信系モデムは、前記送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置を含み、

前記スクランブル符号生成装置は、

所定の生成多項式によるスクランブル符号の系列を生成するために帰還処理および拡散処理を実行するように接続された複数段のレジスタからなるシフトレジスタと、

所定の初期値をもとに前記シフトレジスタを、増加する所定回数だけそれぞれシフト動作させたならば得られる、前記帰還処理および前記拡散処理にそれぞれ関与する前記レジスタの値を演算する演算手段と、

前記演算されたレジスタの値を、対応する前記レジスタに入力する入力手段と

前記演算されかつ入力された値に基づいて前記複数段のレジスタのすべてが有効な値を保持するまで、前記演算手段が前記レジスタの値の演算を行ないかつ前記入力手段が演算された値をレジスタに入力するように前記演算手段および前記入力手段を制御する制御手段とを含み、

前記シフトレジスタは、前記複数段のレジスタのすべてに保持された有効な値をもとにシフト動作を続行して前記スクランブル符号の系列を生成する、携帯無線端末。

【請求項 1 8】 前記スクランブル符号生成装置は、

前記所定の初期値を保持した記憶手段と、

前記所定の生成多項式に基づき前記増加する所定回数だけそれぞれシフト動作させた後における前記帰還処理および前記拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を供給する行列式供給手段とをさらに含み、

前記演算手段は、前記記憶手段に保持されている前記所定の初期値と前記行列式供給手段から供給される前記行列式とを乗算することにより、前記レジスタの値を演算する、請求項 1 7 に記載の携帯無線端末。

【請求項 1 9】 前記拡スクランブル符号生成装置は、

前記所定の初期値を保持した記憶手段をさらに含み、

前記演算手段は、前記所定の生成多項式に基づき前記増加する所定回数だけそ

れぞれシフト動作させた後における前記帰還処理および前記拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を所定の演算により求め、前記記憶手段に保持されている前記所定の初期値と乗算することにより前記レジスタの値を演算する、請求項 1 7 に記載の携帯無線端末。

【請求項 2 0】 デジタル無線通信の携帯無線端末であって、

送信データを変調する送信系モデムと、

前記送信系モデムの送信データを無線処理して送信無線信号として送出する無線処理装置とを備え、

前記送信系モデムは、前記送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置を含み、

前記スクランブル符号生成装置は、

所定の初期値を保持した記憶手段と、

所定の生成多項式に基づき前記スクランブル符号の系列を構成する各符号の値を決定するための行列式を所定の演算により求める論理手段と、

前記記憶手段に保持されている前記所定の初期値と、前記求められた行列式とを乗算することにより前記スクランブル符号の系列を構成する各符号の値を演算する演算手段とを含む、携帯無線端末。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

この発明は、直交符号生成装置、スクランブル符号生成装置、およびそれらを用いた携帯無線端末に関し、より特定的には、デジタル無線通信において、送信データまたは受信データの拡散処理または逆拡散処理に用いる直交符号を生成する直交符号生成装置、送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置、およびそのような符号生成装置を含む携帯無線端末に関する。

【0 0 0 2】

【従来技術】

近年、移動体通信システムにおいて使用されている CDMA (Code Division

Multiple Access) 方式のようなデジタル無線通信方式では、以下に説明するような 2 重拡散方式が採用されている。

【0 0 0 3】

携帯無線端末から基地局へデータを送信する上りリンクに関しては、携帯無線端末において、次のような 2 段階の処理が行なわれる。まず、送信すべきデジタルデータ（シンボルデータ）は、直交符号により拡散処理され、次に、スクランブル符号によりスクランブル処理される。

【0 0 0 4】

送信データ（シンボルデータ）を直交符号により拡散処理するのは、より多くのデータを送信するためである。すなわち、端末において複数のチャネルのシンボルデータを多重化して 1 つのチャネルで基地局へ送信しようとする場合、複数のチャネルのシンボルデータを対応する複数の直交符号系列（シンボルより短い時間幅の符号の系列）でそれぞれ拡散処理した後、得られた複数チャネルの送信データ（チップデータ）を多重化すれば、1 つのチャネルの最大伝送レートを超える高速レートで複数チャネルのデータの伝送が可能となる。

【0 0 0 5】

このようにして得られた送信データ（チップデータ）は、ユーザ（端末）を識別するためにユーザに固有のスクランブル符号によってスクランブル処理されて、基地局に送信される。

【0 0 0 6】

一方、基地局から携帯無線端末へデータを送信する下りリンクに関しては、基地局において、次のような 2 段階の処理が行なわれる。まず、送信すべきデジタルデータ（シンボルデータ）は、直交符号により拡散処理され、次に、スクランブル符号によりスクランブル処理される。

【0 0 0 7】

すなわち、チャネルを識別するため、送信データ（シンボルデータ）をチャネルごとの直交符号系列（シンボルより短い時間幅の符号の系列）により拡散処理する。

【0 0 0 8】

このようにして得られた送信データ（チップ）は、セル（基地局）を識別するためにセルに固有のスクランブル符号によってスクランブル処理されて、携帯無線端末に送信される。

#### 【 0 0 0 9 】

以上のように、上りリンクおよび下りリンクにおいて、それぞれ目的は異なるものの、直交符号およびスクランブル符号という 2 種類の符号を用いて、拡散処理およびスクランブル処理の 2 段階の処理を行なっているため、2 重拡散方式と称される。

#### 【 0 0 1 0 】

このような 2 重拡散方式が採用されたデジタル無線通信の携帯無線端末では、具体的に次のような構成が必要となる。

#### 【 0 0 1 1 】

すなわち、基地局へのデータの送信のためには、送信データ（シンボルデータ）の拡散処理のために直交符号の系列を発生する直交符号生成装置が必要であり、送信データ（チップデータ）のスクランブル処理のためにスクランブル符号の系列を発生するスクランブル符号生成装置が必要である。

#### 【 0 0 1 2 】

次に、基地局において上述の拡散処理およびスクランブル処理が施されたデータの受信のためには、受信データ（チップデータ）のデスクランブル処理のためにスクランブル符号の系列を発生するスクランブル符号生成装置が必要であり、受信データ（チップデータ）の逆拡散処理のために直交符号の系列を発生する直交符号生成装置が必要である。

#### 【 0 0 1 3 】

送信データの拡散処理および受信データの逆拡散処理には同じ種類の直交符号生成装置を用いることができるが、送信データのスクランブル処理および受信データのデスクランブル処理には、回路構成上の理由で異なる種類のスクランブル符号生成装置が用いられる。

#### 【 0 0 1 4 】

以下に、携帯無線端末において送受信データの拡散処理に用いられる直交符号

生成装置について説明する。

【 0 0 1 5 】

まず、直交符号生成装置では、各々が複数ビットからなる（複数ビット周期の）、複数の直交符号系列をすべてメモリに予め格納しておき、発生すべき直交符号系列を指定する系列番号と、指定された直交符号系列の発生すべき直交符号の位置を指定する位置番号とに基づいて、このメモリからデータを読み出すべきアドレスを発生し、発生したアドレスのデータをメモリから読み出して直交符号として出力するように構成されている。

【 0 0 1 6 】

そして、出力された直交符号が送信デジタルデータ（シンボルデータ）と乗算（排他的論理和演算）されることにより、送信デジタルデータが拡散処理されることになる。

【 0 0 1 7 】

図 1 4 は、従来の直交符号生成装置の一例を示すブロック図である。図 1 4 に示す直交符号生成装置は、基本的に、アドレス発生回路 1 0 a と、メモリ 1 0 b と、パラレルーシリアル変換回路 1 0 c とを備えている。

【 0 0 1 8 】

図 1 4 を参照して、メモリ 1 0 b には、各々が複数ビット周期（たとえば CDMA の規格では、最大 2 5 6 ビット周期の範囲内で可変）の、複数の直交符号系列が予め格納されている。

【 0 0 1 9 】

これらの直交符号系列のうち送信デジタルデータと拡散処理されるべき直交符号系列を指定する系列番号が、図示しない制御回路からアドレス発生回路 1 0 a に与えられる。指定された複数ビット周期の直交符号系列は通常、数ビット単位（たとえば 4 ビット単位）のデータに分割されてメモリ 1 0 b から読み出されるため、各データの先頭ビット位置を特定する位置番号が、同様に図示しない制御回路からアドレス発生回路 1 0 a に与えられる。

【 0 0 2 0 】

アドレス発生回路 1 0 a は、与えられた系列番号と位置番号とに基づいて、指

定された直交符号系列のうち特定された数ビットのデータをメモリ 1 0 b から読み出すためのアドレスおよびメモリアクセス制御信号を発生し、メモリ 1 0 b に与える。

#### 【 0 0 2 1 】

メモリ 1 0 b からは、与えられたアドレスによって特定された直交符号系列のうちの数ビットのデータがまとめて読み出され、パラレルーシリアル変換回路 1 0 c に与えられる。パラレルーシリアル変換回路 1 0 c は、メモリ 1 0 b から読み出された数ビットのデータを直列のビット列に変換し、直交符号系列として出力する。出力された直交符号系列は、送信データとの拡散処理に供される。

#### 【 0 0 2 2 】

次に、携帯無線端末において送信データのスクランブル処理に用いられるスクランブル符号生成装置について説明する。

#### 【 0 0 2 3 】

このスクランブル符号生成装置では、基本的に、当該端末に固有のある初期値が、所定の生成多項式に基づいて構成されたシフトレジスタに設定され、受信側である基地局におけるデスクランブル用のスクランブル符号生成とのタイミングをとるために、初期値をもとに所定回数シフト動作させた後の符号から順次連続して、スクランブル符号系列として出力していくように構成されている。

#### 【 0 0 2 4 】

そして、出力されたスクランブル符号系列が送信デジタルデータ（チップデータ）と乗算されることにより、送信デジタルデータがスクランブル処理されることになる。

#### 【 0 0 2 5 】

図 1 5 は、従来のスクランブル符号生成装置の一例を示すブロック図である。図 1 5 は特に、次の生成多項式によるスクランブル符号系列を生成するように構成されたスクランブル符号生成装置を示すものである。

#### 【 0 0 2 6 】

$$f(X) = X^4 + X^2 + 1$$

図 1 5 に示すスクランブル符号生成装置は、基本的に、縦続接続された 4 段の

レジスタ11, 12, 13, 14からなるシフトレジスタを備えおり、それぞれのレジスタの前段には、対応するセクタ15, 16, 17, 18が設けられている。

#### 【0027】

初期値バッファ19には、後述する初期値 $Ri_3$ ,  $Ri_2$ ,  $Ri_1$ ,  $Ri_0$ が保持されている。なお、初期値バッファ19は、たとえばメモリ、レジスタなどで構成される。

#### 【0028】

セクタ15~18の各々は、基本的に、図示しない制御回路から与えられる制御信号に応じて、前段のレジスタからの出力 $Rs$ または初期値バッファ19に格納されている対応する初期値 $Ri$ のいずれかを選択して次段のレジスタのD入力に与える。

#### 【0029】

より詳細には、セクタ15は、排他的論理和回路20からの出力または初期値 $Ri_3$ のいずれかを選択して、入力 $D_3$ としてレジスタ11にセットする。

#### 【0030】

セクタ16は、前段のレジスタ11の出力 $Rs_3$ または初期値 $Ri_2$ のいずれかを選択して、入力 $D_2$ としてレジスタ12にセットする。レジスタ12の出力 $Rs_2$ はセクタ17の一方入力に与えられるとともに、排他的論理和回路20の一方入力に与えられ帰還処理される。

#### 【0031】

セクタ17は、前段のレジスタ12の出力 $Rs_2$ または初期値 $Ri_1$ のいずれかを選択して、入力 $D_1$ としてレジスタ13にセットする。

#### 【0032】

セクタ18は、前段のレジスタ13の出力 $Rs_1$ または初期値 $Ri_0$ のいずれかを選択して、入力 $D_0$ としてレジスタ14にセットする。レジスタ14の出力 $Rs_0$ はスクランブル符号としてシフトレジスタから出力（拡散処理）されるとともに、排他的論理和回路20の他方入力に与えられて帰還処理される。

#### 【0033】

次に、図 1 5 に示したスクランブル符号生成装置によるスクランブル符号生成動作について説明する。なお、現実のデジタル無線通信（たとえば近年開発されている W (Wideband) - C D M A 方式のデジタル無線通信）の規格では、2 5 段のレジスタによってシフトレジスタが構成されるが、図示の都合上、以後の説明では 4 段のレジスタからなる簡略化されたシフトレジスタを用いて説明するものとする。

## 【 0 0 3 4 】

まず、初期値バッファ 1 9 は、当該端末に固有の初期値  $R i_3$ ,  $R i_2$ ,  $R i_1$ ,  $R i_0$ （すべて 0 の場合を除く）が設定され、4 段のレジスタ 1 1 ~ 1 4 のそれぞれに対応するセクタ 1 5 ~ 1 8 を介してロードされる。その後、ロードされた初期値をもとにシフト動作させながら、レジスタ 1 4 の出力からスクランブル符号  $R s_0$  を順次出力する。そして、レジスタ 1 1 ~ 1 4 のそれぞれには、1 回のシフト動作ごとに次式で表わされる入力  $D_3$ ,  $D_2$ ,  $D_1$ ,  $D_0$  がセットされる。

## 【 0 0 3 5 】

## 【数 1】

$$D_3 \leftarrow R s_2 \oplus R s_0$$

$$D_2 \leftarrow R s_3$$

$$D_1 \leftarrow R s_2$$

$$D_0 \leftarrow R s_1$$

## 【 0 0 3 6 】

この動作を行列を用いて表現すると次式のようなになる。ただし、以下の演算（行列表演算も含む）に用いられる加算は modulo 2 加算とする。

## 【 0 0 3 7 】



【数 2】

$$\begin{bmatrix} D_3 \\ D_2 \\ D_1 \\ D_0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{S3} \\ R_{S2} \\ R_{S1} \\ R_{S0} \end{bmatrix}$$

【0 0 3 8】

上述の4つの初期値がレジスタ11～14のそれぞれにロードされてから、1回シフトされた後のレジスタ11～14のそれぞれの値は、次式の行列式で表わされる。

【0 0 3 9】

【数 3】

$$\begin{bmatrix} D_3 \\ D_2 \\ D_1 \\ D_0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{i3} \\ R_{i2} \\ R_{i1} \\ R_{i0} \end{bmatrix}$$

【0 0 4 0】

また、2回シフトされた後のレジスタ11～14のそれぞれの値は、次式の行列式で表わされる。

【0 0 4 1】

【数 4】

$$\begin{bmatrix} D_3 \\ D_2 \\ D_1 \\ D_0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{i3} \\ R_{i2} \\ R_{i1} \\ R_{i0} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} R_{i3} \\ R_{i2} \\ R_{i1} \\ R_{i0} \end{bmatrix}$$

【0 0 4 2】

一般に、初期値がレジスタ11～14のそれぞれにロードされてから、t回シ

フトされた後のレジスタ11～14のそれぞれの値 $D_3(t)$ ,  $D_2(t)$ ,  $D_1(t)$ ,  $D_0(t)$ は、次式の行列式で表わされる。

【0043】

【数5】

$$\begin{bmatrix} D_3(t) \\ D_2(t) \\ D_1(t) \\ D_0(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}^t \begin{bmatrix} Ri_3 \\ Ri_2 \\ Ri_1 \\ Ri_0 \end{bmatrix}$$

【0044】

以上のように、初期値のレジスタ11～14へのロードからスタートして、シフト動作を繰返すことにより、レジスタ14にセットされる値が決まり、符号 $Rs_0$ が順次生成されることになる。

【0045】

ただし、現実のデジタル無線通信（たとえばW-CDMA）の規格では、初期値からスタートしてシフト動作ごとに連続的に生成される符号をそのままスクランブル符号として用いるわけではない。前述のように、受信側とのタイミング合せの目的で、たとえばレジスタ11～14への初期値のロードから $t$ 回（たとえば100回）シフト後以降のレジスタ値をスクランブル符号として出力するように規格で決められている。言い換えると、初期値のロードから100回シフトするまでスクランブル符号は生成されないことになる。しかしながら100回シフトするには長時間を要する（特にシフトレジスタの段数が多い場合）。

【0046】

ここで、上述の $t$ 回シフト後のレジスタ値を求める行列式を用いれば、実際にシフトレジスタをシフトさせなくても、計算で $t$ 回シフト後のレジスタ値を求めることができる。

【0047】

たとえば、デジタル無線通信の規格で、初期値のロードから100回シフト後のレジスタ値からスクランブル符号系列の生成を開始することが定められてい

ば、次式を計算することにより、100回シフト後におけるレジスタ11～14の値 $D_{3(100)}$ 、 $D_{2(100)}$ 、 $D_{1(100)}$ 、 $D_{0(100)}$ を直ちに求めることができる。

【0048】

【数6】

$$\begin{bmatrix} D_{3(100)} \\ D_{2(100)} \\ D_{1(100)} \\ D_{0(100)} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}^{100} \begin{bmatrix} Ri_3 \\ Ri_2 \\ Ri_1 \\ Ri_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} Ri_3 \\ Ri_2 \\ Ri_1 \\ Ri_0 \end{bmatrix}$$

【0049】

この式における100回シフト後の各レジスタ値を決定する行列は、次式に示すように各行ごとのM行列 $M_{3(100)}$ 、 $M_{2(100)}$ 、 $M_{1(100)}$ 、 $M_{0(100)}$ に分けることができる。

【0050】

【数7】

$$\begin{aligned} M_{3(100)} &= \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \\ M_{2(100)} &= \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \\ M_{1(100)} &= \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \\ M_{0(100)} &= \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix} \end{aligned}$$

【0051】

これらの行列を予め求めて保持しておき、ユーザごとの初期値が与えられれば、両者の演算により、直ちに100回シフト後におけるレジスタ11～14のそれぞれの値を算出することができる。そして、以後シフトレジスタのシフト動作を続行すれば、初期値のロードから100回シフト後以降のスクランブル符号系列が順次生成されることになる。

【0052】

図16は、上述のように、初期値のロードからたとえば100回シフト後のレジスタ値を直ちに決定してスクランブル符号系列の生成を開始することができる

スクランブル符号生成装置を示すブロック図である。

【 0 0 5 3 】

図 1 6 において、レジスタ 1 1 ~ 1 4、セクタ 1 5 ~ 1 8、および排他的論理和回路 2 0 からなるシフトレジスタの構成は、図 1 5 に示したシフトレジスタの構成と同じであり、その説明は繰返さない。

【 0 0 5 4 】

図 1 6 の回路構成において、演算回路 2 1 は、上述の 1 0 0 回シフト後の各レジスタ値を決定するための M 行列  $M_{3(100)}$ 、 $M_{2(100)}$ 、 $M_{1(100)}$ 、 $M_{0(100)}$  が格納されたメモリまたはレジスタを備えており、あるいは、これらの M 行列を発生するように組合された論理ゲートでハードウェア的に構成されている。一方、初期値バッファ 2 2 には、当該端末に固有の初期値  $R_i = [R_{i3}, R_{i2}, R_{i1}, R_{i0}]$  が与えられる。なお、初期値バッファ 2 2 は、たとえばメモリ、レジスタなどで構成される。

【 0 0 5 5 】

演算回路 2 1 は、上述の M 行列式と、初期値バッファ 2 2 に与えられた初期値  $R_i$  との乗算を行ない、1 0 0 回シフト後におけるレジスタ 1 ~ 4 の値  $R_{o3}$ 、 $R_{o2}$ 、 $R_{o1}$ 、 $R_{o0}$  を算出する。

【 0 0 5 6 】

このようにして求めたレジスタ値を、所定のタイミングでセクタ 1 5 ~ 1 8 を介して対応するレジスタ 1 1 ~ 1 4 にロードし、それ以降はシフトレジスタをシフト動作させることにより、1 0 0 回シフト後以降のレジスタ 1 4 の出力符号  $R_{s0}$  をスクランブル符号系列として順次出力することができる。出力されたスクランブル符号系列は、送信データとのスクランブル処理に供される。

【 0 0 5 7 】

【発明が解決しようとする課題】

ところで、図 1 4 に示したような従来の直交符号生成装置では、指定され得るすべての直交符号系列がメモリ 1 0 b に格納される。メモリ 1 0 b は、たとえば 1 6 ビット周期の直交符号系列を 1 6 系列格納する場合には、ビット周期の 2 乗すなわち  $16 \times 16$  ビットの記憶容量が必要となる。

## 【0058】

このように、必要なメモリ容量は通常、直交符号系列のビット周期の2乗に対応するビット数となるため、格納すべき直交符号系列のビット周期が長くなると、必要なメモリ容量は飛躍的に増大することになり、直交符号生成装置の回路規模の著しい増大を招くことになる。

## 【0059】

一方、図16に示した従来のスクランブル符号生成装置では、演算回路21の回路規模が大きくなるという問題がある。図16においては、前述のように説明の都合上、4段のレジスタからなるシフトレジスタを用いた例を示しているが、現実の規格や仕様では、たとえば25段という多段のシフトレジスタが用いられ、このようにシフトレジスタの段数が大きくなるほど演算回路11の回路規模ははるかに大きくなってしまう。

## 【0060】

それゆえに、この発明の目的は、回路規模の小型化を実現した直交符号生成装置を提供することである。

## 【0061】

この発明の他の目的は、回路規模の小型化を実現したスクランブル符号生成装置を提供することである。

## 【0062】

この発明のさらに他の目的は、回路規模の小型化を実現した直交符号生成装置を含む携帯無線端末を提供することである。

## 【0063】

この発明のさらに他の目的は、回路規模の小型化を実現したスクランブル符号生成装置を含む携帯無線端末を提供することである。

## 【0064】

## 【課題を解決するための手段】

この発明によるデータの拡散処理に用いる直交符号を生成する直交符号生成装置は、系列番号および位置番号を供給する手段と、第1の論理演算手段と、第2の論理演算手段と、選択手段とを備える。系列番号および位置番号を供給する手

段は、所定の直交符号発生規則に基づいて定義される複数の直交符号系列のうち、発生すべき直交符号系列を指定する  $n$  ( $n$  は正の整数) ビットの系列番号と、指定された直交符号系列の順次発生すべき直交符号の位置を指定する  $n$  ビットの位置番号とを供給する。第 1 の論理演算手段は、系列番号の最下位ビットと位置番号の最下位ビットとを順次論理演算することにより要素符号を発生する。最下位ビット同士の論理演算により発生し得る要素符号の組合せとその組合せを反転させたものとを、要素符号の組合せのパターンに基づいて繰返し配列することにより複数の直交符号系列全体が定義される。第 2 の論理演算手段は、系列番号の最下位ビットを除く  $(n-1)$  ビットと位置番号の最下位ビットを除く  $(n-1)$  ビットとを順次論理演算する。選択手段は、第 2 の論理演算手段の演算結果に対応して、第 1 の論理演算手段によって発生した要素符号または発生した要素符号を反転させた符号のいずれかを、組合せの配列に基づいて選択して直交符号として出力する。

## 【0065】

好ましくは、第 1 の前記論理演算手段は、系列番号の最下位ビットと位置番号の最下位ビットとの論理積を順次演算して要素符号として発生する論理積回路を含み、発生し得る要素符号の組合せは 3 個の符号「0」と 1 個の符号「1」とを含み、組合せを反転させたものは 3 個の符号「1」と 1 個の符号「0」とを含む。

## 【0066】

好ましくは、第 2 の論理演算手段は、系列番号の最下位ビットを除く  $(n-1)$  ビットと位置番号の最下位ビットを除く  $(n-1)$  ビットとの間で、対応する桁の値がともに 1 である桁の個数が 0 または偶数であるか、あるいは奇数であるかを判定する判定手段を含み、選択手段は、判定手段による判定結果が 0 または偶数の場合に論理積回路によって発生した要素符号を直交符号として選択し、判定結果が奇数の場合に要素符号を反転させた符号を直交符号として選択する。

## 【0067】

好ましくは、判定手段は、系列番号の最下位ビットを除く  $(n-1)$  ビットと位置番号の最下位ビットを除く  $(n-1)$  ビットとの間で、対応する桁同士の論

理積を演算する  $(n-1)$  個の論理積回路と、 $(n-1)$  個の論理積回路の演算結果を加算して、その結果が 0 または偶数であるか、あるいは奇数であるかを示す制御信号を発生する加算回路とを含む。

## 【 0 0 6 8 】

好ましくは、直交符号生成装置は、供給される  $n$  ビットの位置番号のうち所定のビットの配列を変更させる位置番号変更手段をさらに備える。

## 【 0 0 6 9 】

好ましくは、要素符号の組合せのパターンに基づく配列の繰返しは周期インデックス  $k$  ( $k$  は正の整数) で規定され、複数の直交符号系列は  $2^k$  個で表わされ、複数の直交符号系列の各々のビット数は  $2^k$  個で表わされ、位置番号変更手段は、 $n$  ビットの位置番号のうち下位の  $k$  ビットの配列を、最下位ビットから  $k$  ビット目までを逆転させるように変更する。

## 【 0 0 7 0 】

この発明の他の局面に従うと、デジタル無線通信の携帯無線端末は、送信および受信のためのデータを処理するモデムと、モデムの送信データを無線処理して送信無線信号として送出し、かつ受信した受信無線信号を無線処理して受信データとしてモデムに与える無線処理装置とを備える。モデムは、データの拡散処理に用いる直交符号を生成する直交符号生成装置を含み、直交符号生成装置は、系列番号および位置番号を供給する手段と、第 1 の論理演算手段と、第 2 の論理演算手段と、選択手段とを含む。系列番号および位置番号を供給する手段は、所定の直交符号発生規則に基づいて定義される複数の直交符号系列のうち、発生すべき直交符号系列を指定する  $n$  ( $n$  は正の整数) ビットの系列番号と、指定された直交符号系列の順次発生すべき直交符号の位置を指定する  $n$  ビットの位置番号とを供給する。第 1 の論理演算手段は、系列番号の最下位ビットと位置番号の最下位ビットとを順次論理演算することにより要素符号を発生する。最下位ビット同士の論理演算により発生し得る要素符号の組合せとその組合せを反転させたものを、要素符号の組合せのパターンに基づいて繰返し配列することにより複数の直交符号系列全体が定義される。第 2 の論理演算手段は、系列番号の最下位ビットを除く  $(n-1)$  ビットと位置番号の最下位ビットを除く  $(n-1)$  ビットと

を順次論理演算する。選択手段は、第 2 の論理演算手段の演算結果に対応して、第 1 の論理演算手段によって発生した要素符号または発生した要素符号を反転させた符号のいずれかを、組合せの配列に基づいて選択して前記直交符号として出力する。

#### 【0071】

好ましくは、第 1 の論理演算手段は、系列番号の最下位ビットと位置番号の最下位ビットとの論理積を順次演算して要素符号として発生する論理積回路を含み、発生し得る要素符号の組合せは 3 個の符号「0」と 1 個の符号「1」とを含み、組合せを反転させたものは 3 個の符号「1」と 1 個の符号「0」とを含む。

#### 【0072】

好ましくは、第 2 の論理演算手段は、系列番号の最下位ビットを除く  $(n-1)$  ビットと位置番号の最下位ビットを除く  $(n-1)$  ビットとの間で、対応する桁の値がともに 1 である桁の個数が 0 または偶数であるか、あるいは奇数であるかを判定する判定手段を含み、選択手段は、判定手段による判定結果が 0 または偶数の場合に論理積回路によって発生した要素符号を直交符号として選択し、判定結果が奇数の場合に要素符号を反転させた符号を直交符号として選択する。

#### 【0073】

好ましくは、判定手段は、系列番号の最下位ビットを除く  $(n-1)$  ビットと位置番号の最下位ビットを除く  $(n-1)$  ビットとの間で、対応する桁同士の論理積を演算する  $(n-1)$  個の論理積回路と、 $(n-1)$  個の論理積回路の演算結果を加算して、その結果が 0 または偶数であるか、あるいは奇数であるかを示す制御信号を発生する加算回路とを含む。

#### 【0074】

好ましくは、直交符号生成装置は、供給される  $n$  ビットの位置番号のうち所定のビットの配列を変更させる位置番号変更手段をさらに備える。

#### 【0075】

好ましくは、要素符号の組合せのパターンに基づく配列の繰返しは周期インデックス  $k$  ( $k$  は正の整数) で規定され、複数の直交符号系列は  $2^k$  個で表わされ、複数の直交符号系列の各々のビット数は  $2^k$  個で表わされ、位置番号変更手段



は、 $n$ ビットの位置番号のうち下位の $k$ ビットの配列を、最下位ビットから $k$ ビット目までを逆転させるように変更する。

#### 【0076】

この発明のさらに他の局面に従うと、送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置は、シフトレジスタと、演算手段と、入力手段と、制御手段とを備える。シフトレジスタは、所定の生成多項式によるスクランブル符号の系列を生成するために帰還処理および拡散処理を実行するように接続された複数段のレジスタからなる。演算手段は、所定の初期値をもとにシフトレジスタを、増加する所定回数だけそれぞれシフト動作させたならば得られる、帰還処理および拡散処理にそれぞれ関与するレジスタの値を演算する。入力手段は、演算されたレジスタの値を、対応するレジスタに入力する。制御手段は、演算されかつ入力された値に基づいて複数段のレジスタのすべてが有効な値を保持するまで、演算手段がレジスタの値の演算を行ないかつ入力手段が演算された値をレジスタに入力するように演算手段および入力手段を制御する。シフトレジスタは、複数段のレジスタのすべてに保持された有効な値をもとにシフト動作を続行してスクランブル符号の系列を生成する。

#### 【0077】

好ましくは、スクランブル符号生成装置は、所定の初期値を保持した記憶手段と、所定の生成多項式に基づき増加する所定回数だけそれぞれシフト動作させた後における帰還処理および拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を供給する行列式供給手段とをさらに備え、演算手段は、記憶手段に保持されている所定の初期値と行列式供給手段から供給される行列式とを乗算することによりレジスタの値を演算する。

#### 【0078】

好ましくは、スクランブル符号生成装置は、所定の初期値を保持した記憶手段をさらに備え、演算手段は、所定の生成多項式に基づき増加する所定回数だけそれぞれシフト動作させた後における帰還処理および拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を所定の演算により求め、記憶手段に保持されている所定の初期値と乗算することによりレジスタの値を演算する。

## 【 0 0 7 9 】

この発明の他の局面に従うと、送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置は、記憶手段と、論理手段と、演算手段とを備える。記憶手段は、所定の初期値を保持する。論理手段は、所定の生成多項式に基づきスクランブル符号の系列を構成する各符号の値を決定するための行列式を所定の演算により求める。演算手段は、記憶手段に保持されている所定の初期値と、求められた行列式とを乗算することによりスクランブル符号の系列を構成する各符号の値を演算する。

## 【 0 0 8 0 】

この発明のさらに他の局面に従うと、デジタル無線通信の携帯無線端末は、送信データを変調する送信系モデムと、送信系モデムの送信データを無線処理して送信無線信号として送出する無線処理装置とを備える。送信系モデムは、送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置を含み、スクランブル符号生成装置は、シフトレジスタと、演算手段と、入力手段と、制御手段とを含む。シフトレジスタは、所定の生成多項式によるスクランブル符号の系列を生成するために帰還処理および拡散処理を実行するように接続された複数段のレジスタからなる。演算手段は、所定の初期値をもとにシフトレジスタを、増加する所定回数だけそれぞれシフト動作させたならば得られる、帰還処理および拡散処理にそれぞれ関与するレジスタの値を演算する。入力手段は、演算されレジスタの値を、対応するレジスタに入力する。制御手段は、演算されたかつ入力された値に基づいて複数段のレジスタのすべてが有効な値を保持するまで、演算手段がレジスタの値の演算を行ないかつ入力手段が演算された値をレジスタに入力するように演算手段および入力手段を制御する。シフトレジスタは、複数段のレジスタのすべてに保持された有効な値をもとにシフト動作を続行してスクランブル符号の系列を生成する。

## 【 0 0 8 1 】

好ましくは、スクランブル符号生成装置は、所定の初期値を保持した記憶手段と、所定の生成多項式に基づき増加する所定回数だけそれぞれシフト動作させた後における帰還処理および拡散処理にそれぞれ関与するレジスタの値を決定する

ための行列式を供給する行列式供給手段とをさらに含み、演算手段は、記憶手段に保持されている所定の初期値と行列式供給手段から供給される行列式とを乗算することによりレジスタの値を演算する。

## 【 0 0 8 2 】

好ましくは、スクランブル符号生成装置は、所定の初期値を保持した記憶手段をさらに含み、演算手段は、所定の生成多項式に基づき増加する所定回数だけそれぞれシフト動作させた後における帰還処理および拡散処理にそれぞれ関与するレジスタの値を決定するための行列式を所定の演算により求め、記憶手段に保持されている所定の初期値と乗算することによりレジスタの値を演算する。

## 【 0 0 8 3 】

この発明のさらに他の局面に従うと、デジタル無線通信の携帯無線端末は、送信データを変調する送信系モデムと、送信系モデムの送信データを無線処理して送信無線信号として送出する無線処理装置とを備える。送信系モデムは、送信データのスクランブル処理に用いるスクランブル符号を生成するスクランブル符号生成装置を含み、スクランブル符号生成装置は、記憶手段と、論理手段と、演算手段とを含む。記憶手段は、所定の初期値を保持する。論理手段は、所定の生成多項式に基づきスクランブル符号の系列を構成する各符号の値を決定するための行列式を所定の演算により求める。演算手段は、記憶手段に保持されている所定の初期値と、求められた行列式とを乗算することによりスクランブル符号の系列を構成する各符号の値を演算する。

## 【 0 0 8 4 】

したがって、この発明によれば、所定の直交符号発生規則に基づいて定義される複数の直交符号系列から、指定された系列番号および位置番号の論理演算の結果によって必要な直交符号を発生するように構成したので、複数の直交符号系列を予め格納するメモリが不要となり、必要な直交符号系列のビット周期が長くなっても直交符号生成装置の回路規模の増大を抑えることができる。

## 【 0 0 8 5 】

さらに、この発明によれば、シフトレジスタを構成する複数段のレジスタのうち、帰還処理および拡散処理に関与するレジスタの値のみを演算手段で演算する

ように構成したので、シフトレジスタの段数が増加した場合でもスクランブル符号生成装置の回路規模の増大を抑えることができる。

## 【 0 0 8 6 】

さらに、この発明によれば、スクランブル符号系列を構成する各符号の値を決定するための行列式を求め、所定の初期値と求められた行列式とを乗算して拡散符号系列を構成する各符号の値を演算するように構成したので、シフトレジスタを用いることなく、スクランブル符号の系列を生成することが可能となる。

## 【 0 0 8 7 】

## 【発明の実施の形態】

以下、この発明の実施の形態を図面を参照して詳しく説明する。なお、図中同一または相当部分には同一符号を付してその説明は繰返さない。

## 【 0 0 8 8 】

図 1 は、この発明による直交符号生成装置およびスクランブル符号生成装置が適用される、デジタル無線通信の携帯無線端末の全体構成を示す概略ブロック図である。

## 【 0 0 8 9 】

図 1 に示す携帯無線端末は、大きくは、アンテナ 1 と、無線処理部 2 と、ベースバンド処理部 3 と、マイクおよびスピーカからなる音声入出力装置 4 と、外部メモリ 5 と、液晶表示装置 (Liquid Crystal Display: 以下、LCD) およびキーからなる表示/入力装置 6 とから構成される。

## 【 0 0 9 0 】

特に、ベースバンド処理部 3 は、モデム 3 a と、チャネルコーデック 3 b と、デジタル信号プロセッサ (Digital Signal Processor: 以下、DSP) 3 c と、中央処理装置 (Central Processing Unit: 以下、CPU) 3 d と、内部メモリ 3 e と、外部インタフェース 3 f と、内部バス 3 g とを含んでいる。

## 【 0 0 9 1 】

アンテナ 1 で受信した、図示しない基地局からの電波信号は、無線処理部 2 によってベースバンドの信号に変換され、ベースバンド処理部 3 に与えられる。

## 【 0 0 9 2 】

ベースバンド処理部 3 において、受信信号はモデム 3 a によって復調され、さらにチャンネルコーデック 3 b によって復号化されて DSP 3 c に与えられる。

【0093】

DSP 3 c は、受信信号をデータ処理して音声入出力装置 4 のスピーカを駆動し、受信信号を音声に変換する。

【0094】

一方、音声入出力装置 4 のマイクで入力された音声は、DSP 3 c でデータ処理され、チャンネルコーデック 3 b に与えられる。チャンネルコーデック 3 b は与えられた音声信号を符号化してモデム 3 a に与え、モデム 3 a は与えられた送信信号を変調して無線処理部 2 に与える。無線処理部 2 は、送信信号に無線処理を施してアンテナ 1 を介して図示しない基地局に向かって送出する。

【0095】

なお、モデム 3 a、チャンネルコーデック 3 b、および DSP 3 c には、内部バス 3 g を介して、CPU 3 d、内部メモリ 3 e、外部インタフェース 3 f とが接続されている。CPU 3 d は、内部メモリ 3 e に格納されているプログラムに従って図 1 の携帯無線端末全体の動作を制御する。また外部インタフェース 3 f は、外部メモリ 5 および表示／入力装置 6 とのインタフェースとして機能する。

【0096】

次に、この発明による直交符号生成装置は、携帯無線端末のうち、送信系のモデムにおいて送信データの拡散処理に用いる直交符号系列を生成するために、または受信系のモデムにおいて受信データの逆拡散処理に用いる直交符号系列を生成するために用いられる。一方、この発明によるスクランブル符号生成装置は、回路構成上の理由で、送信系のモデムにおいて送信データのスクランブル処理に用いるスクランブル符号系列を生成するために用いられる。

【0097】

したがって、図 1 のモデム 3 a のうち、この発明による直交符号生成装置およびスクランブル符号生成装置がともに適用される送信系のモデムの構成について以下に説明することとし、直交符号生成装置のみが適用される受信系のモデムの構成についての説明はここでは省略する。

## 【0098】

なお、この発明は直交符号生成装置そのものの発明であり、受信系のモデムの説明は省略するが、この発明による直交符号生成装置が携帯無線端末の受信系にも適用されることは言うまでもない。

## 【0099】

図2は、図1のベースバンド処理部3のモデム3aのうち、上述のように送信系のモデム部30を抽出して示す概略ブロック図である。

## 【0100】

図1のチャネルコーデック3bから出力された送信データは、無線フレーム組立て部30cにおいて無線フレームへと組立てられ、拡散変調部30bに与えられて、前述の2重拡散方式による拡散変調処理が施される。

## 【0101】

より具体的には、拡散変調部30bにおいて、図示しないこの発明による直交符号生成装置によって生成される直交符号が送信データに乗算されて送信データの拡散処理が行なわれる。次に、図示しないこの発明によるスクランブル符号生成装置によって生成されるスクランブル符号が送信データに乗算されて送信データのスクランブル処理が行なわれることになる。

## 【0102】

拡散変調部30bにより拡散変調された送信データは、波形整形部30aに与えられ、図示しないデジタルフィルタを用いて伝送占有帯域を制限する波形整形が施される。波形整形部30aで波形整形された送信データは、図1の無線処理部2に与えられる。

## 【0103】

次に、図2の拡散変調部30bで用いられる、この発明の直交符号生成装置について詳細に説明する。

## 【0104】

まず、この発明の直交符号生成装置の具体的な実施の形態について説明する前に、この発明による直交符号の発生原理について説明する。

## 【0105】

図3は、この発明による直交符号の発生規則の一例を説明する図である。図3を参照して、発生すべき直交符号系列のビット周期は、所定の規則性（周期性）、すなわち後述する要素符号の組合せのパターンに基づいて、符号の組合せを繰返し配列することにより増大する。その繰返しの指標を周期インデックス  $k$  で表わす。この規則によれば一般的に、周期インデックスが  $k$  のとき、 $2^k$  種類の直交符号系列が定義され、各系列は、 $2^k$  のビット周期を有する。

#### 【0106】

より具体的に、図3において、周期インデックス  $k$  が0のときには、発生する全直交符号を表わすマトリクス  $H_0$  は、1つの符号0のみからなる。

#### 【0107】

次に、周期インデックス  $k$  が1のときには、発生する全直交符号を表わすマトリクス  $H_1$  は、 $k=0$  の場合の  $H_0$  を用いて定義され、2つの  $H_0$  からなる行と、1つの  $H_0$  とそれを反転したものとからなる行とで構成される。具体的には、 $H_1$  は、2系統の直交符号系列  $[00]$  と  $[01]$  とから構成される。

#### 【0108】

次に、周期インデックス  $k$  が2のときには、発生する全直交符号を表わすマトリクス  $H_2$  は、 $k=1$  の場合の  $H_1$  を用いて定義され、2つの  $H_1$  からなる行と、1つの  $H_1$  とそれを反転したものとからなる行とで構成される。具体的には、 $H_2$  は、4系統の直交符号系列  $[0000]$  と  $[0101]$  と  $[0011]$  と  $[0110]$  とから構成される。

#### 【0109】

同様にして、周期インデックス  $k$  が1増加するごとに、1つ下位の周期インデックスの直交符号マトリクスから、新たな直交符号マトリクスが定義される。

#### 【0110】

すなわち、この関係を一般化すると、図3の最下段に示すように、周期インデックス  $k$  のときには、発生する全直交符号を表わすマトリクス  $H_k$  は、 $k=k-1$  の場合の  $H_{k-1}$  を用いて定義され、2つの  $H_{k-1}$  からなる行と、1つの  $H_{k-1}$  とそれを反転したものとからなる行とで構成される。

#### 【0111】

図4は、具体的に、周期インデックス  $k$  が4の場合の発生し得るすべての直交符号を示す16行×16列のマトリクス  $H_4$  を示す図である。このマトリクスの16行の各々が、発生し得る直交符号系列を表わしており、各直交符号系列は、0～15の系列番号で識別される。この0～15の系列番号は、4ビットの系列番号信号Bによって表わされる。

【0112】

また、このマトリクスの16列は、各直交符号系列が16ビットのビット周期を有することを示し、各直交符号系列のビット位置は、0～15の位置番号で識別される。この0～15の位置番号は、4ビットの位置番号信号Cによって表わされる。

【0113】

図4の直交符号のマトリクスにおいて、4ビットの系列番号信号Bの最下位ビットは、0または1である。また、4ビットの位置番号信号Cの最下位ビットも、0または1である。

【0114】

図5は、これらの最下位ビット同士を論理積演算した場合の発生し得る結果を示すマトリクスを表わしており、この2行×2列のマトリクスを構成する符号である3個の「0」と1個の「1」とを要素符号と称する。そしてこのマトリクスは、図3に示す周期インデックス  $k$  が1の場合のマトリクス  $H_1$  に対応している。

【0115】

図3に戻って、周期インデックス  $k$  が1の場合と2の場合との対比から明らかのように、 $k=1$  の2行×2列の要素符号のマトリクス  $H_1$  を繰返しの単位として、要素符号が「1」の位置に相当する2行目の2列目においてのみマトリクス  $H_1$  を反転することにより、 $k=2$  のマトリクス  $H_2$  が構成される。

【0116】

以下同様にして、周期インデックス  $k$  が増えるごとに、 $k$  が1つ下位の場合のマトリクス  $H_{k-1}$  を繰返しの単位として、要素符号が「1」の位置に相当する2行目の2列目においてのみマトリクス  $H_{k-1}$  を反転することにより、マトリクス



$H_k$ が構成される。

【0 1 1 7】

図6は、図4に示した周期インデックス $k$ が4の場合の直交符号の全体のマトリクスを、周期インデックス $k$ が1の場合の要素符号のマトリクス $H_1$ およびそれを反転したものを単位として区切って示したものである。縦軸は、4ビットの系列番号信号Bのうち最下位ビット（0または1）を除いた残りの3ビットで特定され、横軸は、4ビットの位置番号信号Cのうち最下位ビット（0または1）を除いた3ビットで特定される。

【0 1 1 8】

この発明は、この図6に示した要素符号のマトリクス $H_1$ およびそれを反転したものの特定の配列から、図4に示す複数系列（16系列）の直交符号を、予めメモリに格納しておくことなく、論理演算によって発生しようとするものである。

【0 1 1 9】

図7は、図6のマトリクスにおいて、対応する3桁の系列番号（最下位ビットを除く）と3桁の位置番号（最下位ビットを除く）とを対比し、対応する桁の値がともに1である桁の個数を表わしたものである。たとえば、3ビットの系列番号「000」と3ビットの位置番号「010」との間で、対応する桁の値がともに1である桁の個数は0である。一方、3ビットの系列番号「001」と3ビットの位置番号「001」との間で、対応する桁の値がともに1である桁の個数は1である。さらに、3ビットの系列番号「011」と3ビットの位置番号「011」との間で、対応する桁の値がともに1である桁の個数は2である。さらに、3ビットの系列番号「111」と3ビットの位置番号「111」との間で、対応する桁の値がともに1である桁の個数は3である。

【0 1 2 0】

この結果を図6の要素符号のマトリクス $H_1$ およびそれを反転したものの特定の配列と対比すると、対応する桁の値がともに1である桁の個数が0および偶数である系列番号および位置番号に対応するマトリクスは $H_1$ であり、対応する桁の値がともに1である桁の個数が奇数である系列番号および位置番号に対応する

マトリクスは $H_1$ を反転したものであることが理解される。

#### 【0 1 2 1】

以上のことから、発生すべき直交符号系列およびそのビット位置が指定されると、そのときの系列番号信号 $B$ の最下位ビットと位置番号信号の最下位ビットとの論理積演算により、マトリクス $H_1$ の構成要素である要素符号が生成され、さらに系列番号信号 $B$ の最下位ビットを除く3ビットと位置番号信号 $C$ の最下位ビットをのぞく3ビットとの組合せ（対応する桁の値がともに1である桁の個数）から、対応する要素符号のマトリクス $H_1$ の反転の有無が判定される。このような論理演算から、たとえば図4に示すような16系統の直交符号系列のうちのいずれをも発生することができる。このような関係は、周期インデックス $k$ が4の場合に限らず、いかなる周期インデックス $k$ に対しても成立する。

#### 【0 1 2 2】

##### 〔実施の形態1〕

図8は、上述の直交符号の発生原理を実現するための、この発明の実施の形態1による直交符号生成装置を示すブロック図である。図8を参照して、図示しない制御回路（たとえば図1のCPU3d）から、発生すべき直交符号系列を指定する $n$ ビットの系列番号信号 $B_0, B_1, B_2, \dots, B_n$ が並列に供給され、発生すべき直交符号位置を指定する $n$ ビットの位置番号信号 $C_0, C_1, C_2, \dots, C_n$ が並列に供給される。

#### 【0 1 2 3】

これらの信号ビットのうち、系列番号の最下位ビット $B_0$ と位置番号の最下位ビット $C_0$ とが論理積回路 $A_0$ に与えられ、それらの論理積が要素符号として出力される。この要素符号は、選択回路41の一方入力に与えられるとともに、インバータ42によって反転されて、選択回路41の他方入力に与えられる。

#### 【0 1 2 4】

系列番号信号の残りの $(n-1)$ ビットと、位置番号信号の残りの $(n-1)$ ビットとは、対応する桁同士が、論理積回路 $A_1, A_2, A_3, \dots, A_n$ に与えられ、それぞれの論理積が出力される。

#### 【0 1 2 5】

すなわち、これらの論理積回路  $A_1, A_2, A_3, \dots, A_n$  の出力は、系列番号と位置番号との対応する桁同士がともに 1 である場合に 1 となり、それ以外の場合には 0 となる。

#### 【 0 1 2 6 】

これらの論理積回路  $A_1, A_2, A_3, \dots, A_n$  の出力は、加算回路 4 0 で、modulo 2 加算される。すなわち、これらの論理積回路  $A_1, A_2, A_3, \dots, A_n$  の出力を加算し、総和を 2 で除算した余りを示す信号が制御信号として選択回路 4 1 の制御入力に与えられる。

#### 【 0 1 2 7 】

したがって、これらの論理積回路  $A_1, A_2, A_3, \dots, A_n$  の出力の総和が 0 または偶数であれば、制御信号は、論理積回路  $A_0$  から対応して出力された要素符号が反転されない要素符号マトリクスに属することを示し、選択回路 4 1 はこの制御信号を受けて、論理積回路  $A_0$  から対応して出力された要素符号を選択して直交符号として出力する。

#### 【 0 1 2 8 】

一方、これらの論理積回路  $A_1, A_2, A_3, \dots, A_n$  の出力の総和が奇数であれば、制御信号は、論理積回路  $A_0$  から対応して出力された要素符号が反転された要素符号マトリクスに属することを示し、選択回路 4 1 はこの制御信号を受けて、論理積回路  $A_0$  から対応して出力された要素符号をインバータ 4 2 で反転したものを選択して直交符号として出力する。

#### 【 0 1 2 9 】

このように、たとえば CPU 3 d から指定された系列番号に対し、位置番号として、増加または減少するカウンタ値を順次与えれば、指定された直交符号系列を構成する直交符号が選択回路 4 1 から 1 ビットずつ順次出力されることになる。

#### 【 0 1 3 0 】

以上のように、この発明の実施の形態 1 によれば、要素符号のマトリクスを周期インデックスを指標に繰返し配列することによって定義される複数の直交符号系列から、指定された系列番号および位置番号の論理積演算の結果によって必要

な直交符号を発生するように構成したので、複数の直交符号系列を予め格納するメモリが不要となり、回路規模の縮小を図ることが可能となる。特に、必要な直交符号系列のビット周期が長くなった場合であっても、単に論理積回路を追加するだけで対応可能であり、直交符号生成装置の回路規模の増大を最小限に抑えることができる。

#### 【0131】

##### 〔実施の形態2〕

図9は、この発明の実施の形態2による直交符号生成装置を示すブロック図である。図9に示した直交符号生成装置は、以下の点を除いて、図8に示した実施の形態1による直交符号生成装置と同じである。

#### 【0132】

すなわち、図9に示した直交符号生成装置では、周期インデックス $k$ に応じて位置番号を変更する位置番号変更回路43が設けられている。この実施の形態2では、図示しない制御回路（たとえばCPU3d内のカウンタ）から与えられる位置番号を位置番号変更回路43によって変更することにより、図4に示した実施の形態1による直交符号とは異なる直交符号を発生することができる。

#### 【0133】

位置番号の変更方法の一例について説明する。周期インデックスが $k$ のとき、前述のように、発生可能な直交符号は、 $2^k$ ビットの符号で構成される。したがって、たとえば図示しないカウンタから位置番号として与えられる所定ビット数のカウンタ値のうち、下位 $k$ ビットによって位置番号が特定される。

#### 【0134】

そこで、位置番号変更回路43において、所定ビット数の位置番号（カウンタ値）が与えられると、周期インデックス $k$ に応じて、位置番号のうちの下位の $k$ ビットの配列を、最下位ビットから $k$ ビット目までを逆転させるように変更する。そうすると、位置番号である入力カウンタ値の増加または減少に応じて、系列番号で指定された直交符号系列の直交符号が1ビットずつ選択回路41から出力されることになる。

#### 【0135】

図 1 0 は、周期インデックス  $k$  が 4 の場合に、図 9 の直交符号生成装置によって発生する直交符号を示す図である。

#### 【 0 1 3 6 】

以上のように、この発明の実施の形態 2 によれば、位置番号変更回路を付加するだけで、直交符号生成装置が発生する直交符号を様々に変化させることができる。

#### 【 0 1 3 7 】

次に、図 2 の拡散変調部 3 0 b で用いられる、この発明のスクランブル符号生成装置について詳細に説明する。

#### 【 0 1 3 8 】

##### 〔実施の形態 3〕

図 1 1 は、この発明の実施の形態 3 によるスクランブル符号生成装置を示すブロック図である。この実施の形態 3 によるスクランブル符号生成装置は、以下の点で、図 1 6 に示した従来のスクランブル符号生成装置と異なっている。

#### 【 0 1 3 9 】

すなわち、図 1 6 の従来のスクランブル符号生成装置では、前述のように、初期値のシフトレジスタへのロードから 1 0 0 回シフト後における各レジスタ値を求めるための  $M$  行列式  $M_3(100)$ ,  $M_2(100)$ ,  $M_1(100)$ ,  $M_0(100)$  を予め求めておき、演算回路 2 1 において、初期値  $R_i$  と乗算することにより、シフトレジスタを構成するレジスタ 1 1 ~ 1 4 のすべての値を算出し、対応するレジスタにロードするように構成していた。

#### 【 0 1 4 0 】

しかしながら、スクランブル符号系列を生成するためには、シフトレジスタを構成するすべてのレジスタの値を求めなくても、そのうち、少なくとも帰還処理および拡散処理に関与するレジスタの値のみを演算すれば足りる。

#### 【 0 1 4 1 】

すなわち、帰還処理および拡散処理に関するレジスタとは、図 1 1 のスクランブル符号生成装置のシフトレジスタを構成するレジスタ 1 1 ~ 1 4 のうち、符号  $R_{s_0}$  をスクランブル符号として出力するとともに排他的論理和回路 2 0 に帰還

処理するレジスタ 1 4 と、符号  $R s_2$  を排他的論理和回路 2 0 に帰還処理するレジスタ 1 2 とを意味している。

#### 【0 1 4 2】

図 1 1 の構成において、ある位相でレジスタ 1 2, 1 4 の値が外部から与えられれば、次の位相、すなわち 1 回のシフト動作後に、排他的論理和回路 2 0 によってレジスタ 1 1 の値  $D_3$  が決定される。また、レジスタ 1 2 の値  $R s_2$  はレジスタ 1 3 にシフトされる。さらに、この位相においてもう 1 度、レジスタ 1 2, 1 4 の値が外部から与えられれば、次の位相において、レジスタ 1 1 ~ 1 4 のすべてが有効な値を保持することになる。その後は、シフトレジスタのシフト動作を続行すればスクランブル符号系列がレジスタ 1 4 から順次出力されることになる。言換えると、本来必要なレジスタの値は、レジスタ 1 2, 1 4 の値のみである。

#### 【0 1 4 3】

したがって、図 1 1 に示す実施の形態 3 によるスクランブル符号生成装置では、レジスタ 1 2, 1 4 に対応する 1 0 0 回シフト後の M 行列  $M_2(100)$ ,  $M_0(100)$ 、および 1 0 1 回シフト後の M 行列  $M_2(101)$ ,  $M_0(101)$  を次式を用いて予め算出し ( $t = 1 0 0, 1 0 1$ )、制御回路 2 4 に含まれるメモリまたはレジスタの記憶領域  $M_2/M_0$  に格納しておく。

#### 【0 1 4 4】

なお、M 行列の供給の方法としては、上述のようにメモリ、レジスタなどの記憶手段に記憶させ、そこから読出す方法の他に、制御回路 2 4 を、これらの M 行列を発生するように組合された論理ゲートでハードウェア的に構成する方法もある。以下に説明する各実施形態では説明の便宜上、M 行列はメモリに格納されることとするが、M 行列の供給方法はこれに限られるものではない。

#### 【0 1 4 5】

【数 8】

$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}^t$$

【0 1 4 6】

一方、初期値バッファ 2 2 には、当該端末に固有の初期値  $R_i = [R_{i_3}, R_{i_2}, R_{i_1}, R_{i_0}]$  が与えられる。なお、各実施形態では、初期値バッファ 2 2 は、たとえばメモリ、レジスタなどで構成される。

【0 1 4 7】

図 1 1 の演算回路 2 3 はまず、1 0 0 回シフト後の符号を出力する位相  $P_{(100)}$  の 1 位相前の  $P_{(99)}$  のタイミングで、制御回路 2 4 内のメモリに格納されている 1 0 0 回シフト後の M 行列  $M_{2(100)}$ ,  $M_{0(100)}$  と、初期値バッファ 2 2 に格納されている初期値  $R_i$  とを乗算して、1 0 0 回シフト後の位相  $P_{(100)}$  におけるレジスタ 1 2, 1 4 への入力  $R_{o_2(100)}$ ,  $R_{o_0(100)}$  を算出する。

【0 1 4 8】

このように算出された値  $R_{o_2(100)}$ ,  $R_{o_0(100)}$  はそれぞれ、制御回路 2 4 による制御下にセクタ 1 6, 1 8 によって選択され、対応するレジスタ 1 2, 1 4 へ入力  $D_2$ ,  $D_0$  として与えられ、1 0 0 回シフト後の符号を出力する位相  $P_{(100)}$  においてラッチされる。それぞれのレジスタ 1 2, 1 4 の出力  $R_{s_2}$ ,  $R_{s_0}$  のうち、 $R_{s_0}$  はスクランブル符号として出力され、 $R_{s_2}$ ,  $R_{s_0}$  は排他的論理和回路 2 0 を介して帰還され、入力  $D_3$  としてレジスタ 1 1 に与えられる。

【0 1 4 9】

したがって、1 0 0 回シフト後の位相  $P_{(100)}$  のタイミングで、レジスタ 1 2, 1 4 は、1 0 0 回シフト後に相当する有効な値を保持し出力していることになるが、この時点ではレジスタ 1 1, 1 3 には有効な値は保持されていないことになる。

【0 1 5 0】

したがって、演算装置 2 3 は、1 0 0 回シフト後の位相  $P_{(100)}$  のタイミングで、制御回路 2 4 内のメモリに格納されている 1 0 1 回シフト後の M 行列  $M_{2(101)}$ 、 $M_{0(101)}$  と、初期値バッファ 2 2 に格納されている初期値  $R_i$  とを乗算して、1 0 1 回シフト後の位相  $P_{(101)}$  におけるレジスタ 1 2, 1 4 への入力値  $R_{o_2(101)}$ 、 $R_{o_0(101)}$  を算出する。

【0 1 5 1】

このように算出された値  $R_{o_2(101)}$ 、 $R_{o_0(101)}$  はそれぞれ、セクタ 1 6, 1 8 によって選択され、対応するレジスタ 1 2, 1 4 へ入力  $D_2$ 、 $D_0$  として与えられ、1 0 1 回シフト後の符号を出力する位相  $P_{(101)}$  においてラッチされる。またこの位相  $P_{(101)}$  において、位相  $P_{(100)}$  においてレジスタ 1 2 から出力されていた値  $R_{s_2}$  はレジスタ 1 3 にラッチされ、かつ排他的論理和回路 2 0 からの値  $D_3$  もレジスタ 1 1 にラッチされる。

【0 1 5 2】

したがって、この時点で、レジスタ 1 1 ~ 1 4 のすべては、1 0 1 回シフト後の位相  $P_{(101)}$  における有効な値を保持していることになる。

【0 1 5 3】

したがって、これ以降、レジスタ 1 1 の出力  $R_{s_3}$  がレジスタ 1 2 の入力に与えられ、レジスタ 1 3 の出力  $R_{s_1}$  がレジスタ 1 4 の入力に与えられるように、セクタ 1 6, 1 8 の選択動作が、制御回路 2 4 内のメモリの記憶領域に保持されているセクタ制御データ  $D_2$ 、 $D_0$  により制御される。

【0 1 5 4】

これにより、位相  $P_{(101)}$  よりも後は、演算回路 2 3 は、レジスタ値を演算する必要はなく、レジスタ 1 1 ~ 1 4 に保持された有効な値をシフトさせることにより、レジスタ 1 4 の出力からスクランブル符号系列が順次生成されることになる。

【0 1 5 5】

以上のように、この発明の実施の形態 3 によれば、シフトレジスタを構成するレジスタのうち、帰還処理および拡散処理に関与するレジスタの値のみを演算するように構成しているので、演算回路の小型化を図ることができる。なお、図 1



1の構成では、図16に示した従来例と比較して、制御回路24が付加されているので、一見したところ回路規模が増大しているかのように見える。しかしながら、シフトレジスタの段数が多くなるほど、図16の従来例の演算回路21の演算量および回路規模は莫大なものとなり、これと対比すると、図11の実施の形態3では、制御回路24の付加に比べて演算回路23の演算量および回路規模の縮小による効果が著しく、全体としてスクランブル符号生成装置の回路規模を著しく縮小することが可能となる。

【0156】

〔実施の形態4〕

図12は、この発明の実施の形態4によるスクランブル符号生成装置を示すブロック図である。この実施の形態4によるスクランブル符号生成装置は、以下の点で、図11に示した実施の形態3によるスクランブル符号生成装置と異なっている。

【0157】

すなわち、図11の実施の形態3のスクランブル符号生成装置では、レジスタ12に対応するM行列 $M_{2(100)}$ 、 $M_{2(101)}$ と、レジスタ14に対応するM行列 $M_{0(100)}$ 、 $M_{0(101)}$ とを予め算出して制御回路24内のメモリに格納しておき、位相 $P_{(99)}$ 、 $P_{(100)}$ の各々ごとにレジスタ値の演算に必要なM行列を選択して抽出していたが、図12の実施の形態4のスクランブル符号生成装置では、これと等価な処理を異なる回路構成で実行している。

【0158】

すなわち、各レジスタの値は、シフトレジスタを介するデータのシフトと帰還処理とによって求められる。したがって、各レジスタの値を求めるためのM行列同士も、互いに時間的、空間的な相関性を有するものである。このような相関性を用いれば、ある1つのM行列から別のM行列を簡単な回路で求めることができる。

【0159】

たとえば、図11の実施の形態3のように4つのM行列を予めメモリに格納しておかなくても、1つのM行列が与えられれば残りの3つのM行列を簡単な回路

構成で算出することができる。以下に説明する実施の形態4では、1つのM行列として $M_{0(100)}$ が与えられた場合に、残りの3つのM行列 $M_{0(101)}$ ,  $M_{2(100)}$ ,  $M_{2(101)}$ を求める回路構成を示している。

【0160】

まず、各レジスタの $t$ 回シフト後の値を求めるためのM行列は、上述の数8の行列の演算結果を各行ごとに分けることによって得られる。したがって、さらに1回シフトした合計 $(t+1)$ 回シフト後におけるM行列は、次式を演算することによって得られる。

【0161】

【数9】

$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}^t \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

【0162】

ここで、たとえば $t$ 回シフト後におけるレジスタ14に対応するM行列を $M_{0(t)}$ で表わすと、このM行列の行列要素は次式のように表わされる。

【0163】

【数10】

$$M_{0(t)} = [M_{0(t,3)} \quad M_{0(t,2)} \quad M_{0(t,1)} \quad M_{0(t,0)}]$$

【0164】

このように表現した場合、 $t$ 回シフト後におけるM行列 $M_{0(t)}$ と、 $(t+1)$ 回シフト後におけるM行列 $M_{0(t+1)}$ との間の時間的關係は次式で表現される。

【0165】

## 【数 1 1】

$$M_{0(t+1,3)} = M_{0(t,2)}$$

$$M_{0(t+1,2)} = M_{0(t,3)} \oplus M_{0(t,1)}$$

$$M_{0(t+1,1)} = M_{0(t,0)}$$

$$M_{0(t+1,0)} = M_{0(t,3)}$$

## 【0 1 6 6】

したがって、上述の実施の形態に戻れば、100回シフト後のレジスタ14のM行列である $M_{0(100)}$ の各行列要素が与えられれば、101回シフト後のレジスタ14のM行列である $M_{0(101)}$ の各行列要素は上式より容易に求まる。

## 【0 1 6 7】

また、レジスタ12の値 $Rs_2$ は2回シフトされてレジスタ14の値 $Rs_0$ となるので、 $t$ 回シフト後におけるレジスタ14のM行列である $M_{0(t)}$ と、 $t$ 回シフト後におけるレジスタ12のM行列である $M_{2(t)}$ との空間的關係は、 $t$ 回シフト後におけるレジスタ14のM行列である $M_{0(t)}$ と $(t+2)$ 回シフト後におけるレジスタ14のM行列である $M_{0(t+2)}$ との時間的關係と等価である。

## 【0 1 6 8】

したがって、数11に示した $M_{0(t+1)}$ と $M_{0(t)}$ との時間的關係を2回適用すれば、 $M_{0(t)}$ を元に $M_{2(t)}$ を求めることができる。

## 【0 1 6 9】

具体的には、次式のように、まず $M_{0(t)}$ から $M_{0(t+1)}$ を求め、さらに $M_{0(t+1)}$ から $M_{0(t+2)}$ を求めれば、結果として $M_{2(t)}$ が求められることになる。

## 【0 1 7 0】

## 【数 1 2】

$$M_{0(t+1,3)} = M_{0(t,2)}$$

$$M_{0(t+1,2)} = M_{0(t,3)} \oplus M_{0(t,1)}$$

$$M_{0(t+1,1)} = M_{0(t,0)}$$

$$M_{0(t+1,0)} = M_{0(t,3)}$$

$$M_{0(t+2,3)} = M_{0(t+1,2)} = M_{0(t,3)} \oplus M_{0(t,1)}$$

$$M_{0(t+2,2)} = M_{0(t+1,3)} \oplus M_{0(t+1,1)} = M_{0(t,2)} \oplus M_{0(t,0)}$$

$$M_{0(t+2,1)} = M_{0(t+1,0)} = M_{0(t,3)}$$

$$M_{0(t+2,0)} = M_{0(t+1,3)} = M_{0(t,2)}$$

## 【0 1 7 1】

図 1 2 の演算回路 2 5 は、このような M 行列の変換を実現するための回路構成である。図 1 2 を参照して、まず、9 9 回シフト後の位相  $P_{(99)}$  において、制御回路 2 6 内のメモリに格納されていた 1 0 0 回シフト後におけるレジスタ 1 4 の M 行列  $M_{0(100)}$  がセレクタ 2 5 a を介して  $M_0$  変換論理回路 2 5 b および乗算回路 2 5 c の一方入力に与えられる。

## 【0 1 7 2】

$M_0$  変換論理回路 2 5 b は、数 1 1 に示した変換論理に基づいて、与えられた M 行列  $M_{0(100)}$  をさらに 1 回シフト後の M 行列  $M_{0(101)}$  に変換してラッチ回路 2 5 d に保持させるとともに、次段の  $M_0$  変換論理回路 2 5 e に与える。

## 【0 1 7 3】

乗算回路 2 5 c は、与えられた M 行列  $M_{0(100)}$  と、初期値バッファ 2 2 に格納されている初期値  $R_i$  との乗算を行ない、レジスタ 1 4 の値である  $R_{00}$  を生成し、セレクタ 1 8 を介してレジスタ 1 4 の入力に与える。

## 【0 1 7 4】

一方、 $M_0$  変換論理回路 2 5 e は、 $M_0$  変換論理回路 2 5 b から出力される  $M_{0(101)}$  を、再度、数 1 1 に示した変換論理に基づいて、さらに 1 回シフト後の  $M_{0(102)}$  すなわち  $M_{2(100)}$  に変換して、乗算回路 2 5 f に与える。乗算回路 2 5 f は、与えられた M 行列  $M_{2(100)}$  と、初期値バッファ 2 2 に格納されている初期値  $R$

i との乗算を行ない、レジスタ 1 2 の値である  $R \circ_2(100)$  を生成し、セクタ 1 6 を介してレジスタ 1 2 の入力に与える。

【 0 1 7 5 】

このように算出された値  $R \circ_2(100)$ 、 $R \circ_0(100)$  はそれぞれ、1 0 0 回シフト後の位相  $P_{(100)}$  においてレジスタ 1 2、1 4 にラッチされる。したがって、位相  $P_{(100)}$  では、レジスタ 1 2、1 4 は 1 0 0 回シフト後の有効な値を保持して出力していることになるが、この時点ではレジスタ 1 1、1 3 には有効な値は保持されていない。

【 0 1 7 6 】

この位相  $P_{(100)}$  のタイミングで、ラッチ回路 2 5 d に保持されている  $M_0(101)$  がセクタ 2 5 a を介して選択され、 $M_0$  変換論理回路 2 5 b および乗算回路 2 5 c に与えられる。

【 0 1 7 7 】

$M_0$  変換論理回路 2 5 b に与えられた  $M_0(101)$  は、数 1 1 の変換論理に基づく変換を受け、さらに  $M_0$  変換論理回路 2 5 e に与えられて再度、数 1 1 の変換論理に基づく変換を受ける。すなわち、 $M_0(101)$  は、2 回の変換論理処理により空間的に  $M_2(101)$  に変換され乗算回路 2 5 f に与えられる。

【 0 1 7 8 】

乗算回路 2 5 c は、与えられた  $M_0(101)$  と、初期値バッファ 2 2 に格納されている初期値  $R_i$  との乗算を行ない、レジスタ 1 4 の値である  $R \circ_0(101)$  を生成し、セクタ 1 8 を介してレジスタ 1 4 の入力に与える。

【 0 1 7 9 】

一方、乗算回路 2 5 f は、与えられた  $M_2(101)$  と、初期値バッファ 2 2 に格納されている初期値  $R_i$  との乗算を行ない、レジスタ 1 2 の値である  $R \circ_2(101)$  を生成し、セクタ 1 6 を介してレジスタ 1 2 の入力に与える。

【 0 1 8 0 】

このように算出された値  $R \circ_2(101)$ 、 $R \circ_0(101)$  はそれぞれ、1 0 1 回シフト後の位相  $P_{(101)}$  においてレジスタ 1 2、1 4 にラッチされる。また、この位相において、位相  $P_{(100)}$  においてレジスタ 1 2 から出力されていた値  $R s_2$  はレジ

スタ 1 3 にラッチされ、排他的論理和回路 2 0 からの値  $D_3$  もレジスタ 1 1 にラッチされる。

【0 1 8 1】

したがって、この時点で、レジスタ 1 1 ~ 1 4 のすべては、1 0 1 回シフト後の位相  $P_{(101)}$  における有効な値を保持していることになる。

【0 1 8 2】

したがって、これ以降、レジスタ 1 1 の出力  $R s_3$  がレジスタ 1 2 に与えられ、レジスタ 1 3 の出力  $R s_1$  がレジスタ 1 4 に与えられるように、セクタ 1 6 , 1 8 の選択動作が、制御回路 2 4 内のメモリの記憶領域に保持されているセクタ制御データにより制御される。

【0 1 8 3】

これにより、位相  $P_{(101)}$  よりも後は、演算回路 2 5 はレジスタ値を演算する必要はなく、レジスタ 1 1 ~ 1 4 に保持されている有効な値をシフトさせることにより、レジスタ 1 4 の出力からスクランブル符号系列が順次得られることになる。

【0 1 8 4】

以上のように、この発明の実施の形態 4 によれば、図 1 1 に示した実施の形態 3 と等価な機能を実現することができるので、シフトレジスタを構成するレジスタのうち帰還処理および拡散処理に関与するレジスタの値のみを演算することにより、演算回路の小型化を図ることができる。特に、前述の実施の形態 3 のように連続する 2 つの位相にわたって合計 4 つの M 行列を記憶する代わりに 1 つの M 行列のみを記憶し、この 1 つの M 行列に基づいて残りの 3 つの M 行列を簡単な回路構成で算出するようにしているので、必要なメモリ容量を削減し、スクランブル符号生成装置全体の回路構成をさらに縮小することができる。

【0 1 8 5】

〔実施の形態 5〕

図 1 3 は、この発明の実施の形態 5 によるスクランブル符号生成装置を示すブロック図である。

【0 1 8 6】

前述の実施の形態 3 および 4 では、演算回路 2 3 または 2 5 によって、レジスタ 1 2, 1 4 の値を生成して対応するレジスタにロードしていき、レジスタ 1 1 ~ 1 4 のすべてが有効なレジスタ値で満たされた後は、シフトレジスタ自体のシフト動作により拡散符号系列を生成していた。

【0 1 8 7】

しかしながら、図 1 2 の実施の形態 4 の演算回路 2 5 に着目すると、その中の乗算回路 2 5 c から出力される値  $R_{o0}$  は、セクタ 1 8 およびレジスタ 1 4 を介して拡散符号  $R_{s0}$  として出力される値である。したがって、乗算回路 2 5 c の出力  $R_{o0}$  を連続して抽出すれば、シフトレジスタを用いることなく、スクランブル符号系列を直接生成することができる。

【0 1 8 8】

図 1 3 の実施の形態 5 によるスクランブル符号生成装置は、図 1 2 の実施の形態 4 における演算回路 2 5 を簡略化してスクランブル符号生成装置として構成したものである。

【0 1 8 9】

図 1 3 を参照して、M 行列バッファ 2 7 に格納されていた t 回シフト後における M 行列  $M_0(t)$  がセクタ 2 8 a によって選択されてラッチ 2 8 b に保持され、その出力が  $M_0$  変換論理回路 2 8 c と乗算回路 2 8 d とに与えられる。なお、前述の各実施形態と同様に、M 行列バッファ 2 7 は、メモリ、レジスタなどの記憶手段で構成してもよく、または M 行列を発生するように組合された論理ゲートからなるハードウェアの回路構成を用いてもよい。

【0 1 9 0】

乗算回路 2 8 d は、与えられた  $M_0(t)$  と、初期値バッファ 2 2 に保持されている初期値  $R_i$  との乗算を行ない、値  $R_{o0}(t)$  を生成する。この値は、スクランブル符号  $R_{s0}(t)$  として出力される。

【0 1 9 1】

一方、 $M_0$  変換論理回路 2 8 c は、数 1 1 に示した変換論理に基づいて  $M_0(t)$  を  $M_0(t+1)$  に変換し、セクタ 2 8 a を介してラッチ 2 8 b に保持させる。ラッチ 2 8 b の出力は、 $M_0$  変換論理回路 2 8 c と乗算回路 2 8 d とに与えられる。

## 【0192】

乗算回路28dは、与えられた $M_{0(t+1)}$ と、初期値バッファ22に保持されている初期値 $R_i$ との乗算を行ない、値 $R_{o_{0(t+1)}}$ を生成する。この値は、スクランブル符号 $R_{s_{0(t+1)}}$ として出力される。

## 【0193】

このように、M行列バッファ27に保持されている1つのM行列 $M_{0(t)}$ を元に各位相ごとに数11の変換論理を繰返し適用することにより、各位相に応じたM行列を $M_0$ 変換論理回路28cで順次生成することができ、乗算回路28dで初期値 $R_i$ と乗算することにより、スクランブル符号 $R_{s_0}$ の系列を生成することができる。

## 【0194】

以上のように、この発明の実施の形態5によれば、シフトレジスタを用いることなく演算回路から直接にスクランブル符号系列を生成するように構成しているので、スクランブル符号生成装置のさらなる小型化が可能となる。

## 【0195】

今回開示された実施の形態はすべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

## 【0196】

## 【発明の効果】

以上のように、この発明によれば、所定の直交符号発生規則に基づいて定義される複数の直交符号系列から、指定された系列番号および位置番号の論理演算の結果によって必要な直交符号を発生するように構成したので、複数の直交符号系列を予め格納するメモリが不要となり、回路規模の縮小を図ることが可能となる。特に、必要な直交符号系列のビット周期が長くなった場合であっても、直交符号生成装置の回路規模の増大を抑えることができる。

## 【0197】

さらに、この発明によれば、シフトレジスタを構成する複数段のレジスタのう



ち、帰還処理および拡散処理に關与するレジスタの値のみを演算回路で演算するように構成したので、演算回路の回路規模を縮小することが可能となる。特に、シフトレジスタの段数が増加した場合であっても、スクランブル符号装置全体の回路規模の増大を抑えることができる。

【0198】

さらに、この発明によれば、シフトレジスタを用いることなく、演算回路によって直接スクランブル符号系列を生成するように構成しているので、スクランブル符号生成装置の回路規模をさらに縮小することが可能となる。

【図面の簡単な説明】

【図1】 この発明による直交符号生成装置およびスクランブル符号生成装置が適用される携帯無線端末の概略ブロック図である。

【図2】 図1に示す携帯無線端末中の送信系モデム部を示す概略ブロック図である。

【図3】 この発明による直交符号の発生規則の一例を説明する図である。

【図4】 この発明の実施の形態1による直交符号生成装置で発生可能な直交符号を示す図である。

【図5】 この発明による要素符号の発生原理を説明する図である。

【図6】 この発明による直交符号を要素符号のマトリクスで表現した図である。

【図7】 図6に示した要素符号のマトリクスの反転状態を数値で表現した図である。

【図8】 この発明の実施の形態1による直交符号生成装置を示す概略ブロック図である。

【図9】 この発明の実施の形態2による直交符号生成装置を示す概略ブロック図である。

【図10】 この発明の実施の形態2による直交符号生成装置で発生可能な直交符号を示す図である。

【図11】 この発明の実施の形態3によるスクランブル符号生成装置を示す概略ブロック図である。

【図 1 2】 この発明の実施の形態 4 によるスクランブル符号生成装置を示す概略ブロック図である。

【図 1 3】 この発明の実施の形態 5 によるスクランブル符号生成装置を示す概略ブロック図である。

【図 1 4】 従来の直交符号生成装置を示す概略ブロック図である。

【図 1 5】 従来のスクランブル符号生成装置を示す概略ブロック図である。

【図 1 6】 従来のスクランブル符号生成装置の他の例を示す概略ブロック図である。

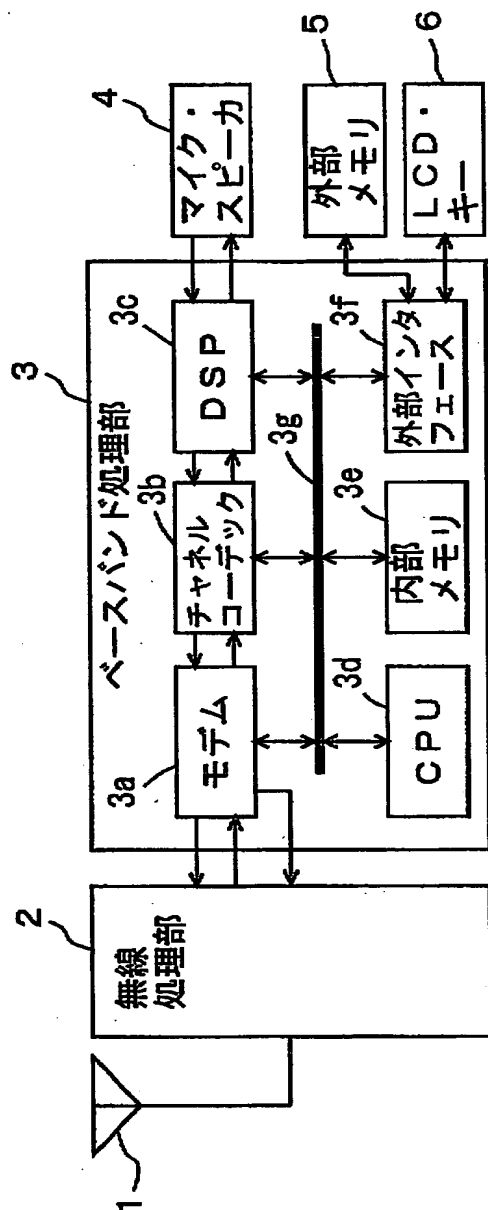
【符号の説明】

1 アンテナ、2 無線処理部、3 ベースバンド処理部、3 a モデム、3 b チャンネルコーデック、3 c DSP、3 d CPU、3 e 内部メモリ、3 f 外部インタフェース、3 g 内部バス、4 音声入出力装置、5 外部メモリ、6 表示／入力装置、10 a アドレス発生回路、10 b メモリ、10 c パラレルーシリアル変換回路、11, 12, 13, 14 レジスタ、15, 16, 17, 18 セレクタ、19, 22 初期値バッファ、20 排他的論理和回路、21, 23, 25, 28 演算回路、24, 26 制御回路、25 a セレクタ、25 b, 25 e  $M_0$ 変換論理回路、25 c, 25 f 乗算回路、25 d ラッチ、27 M行列バッファ、28 a セレクタ、28 b ラッチ、28 c  $M_0$ 変換論理回路、28 d 乗算回路、30 送信系モデム部、30 a 波形整形部、30 b 拡散変調部、30 c 無線フレーム組立部、40 加算回路、41 選択回路、42 インバータ、43 位置番号変更回路、 $A_0, A_1, A_2, A_3, \dots, A_n$  論理積回路。

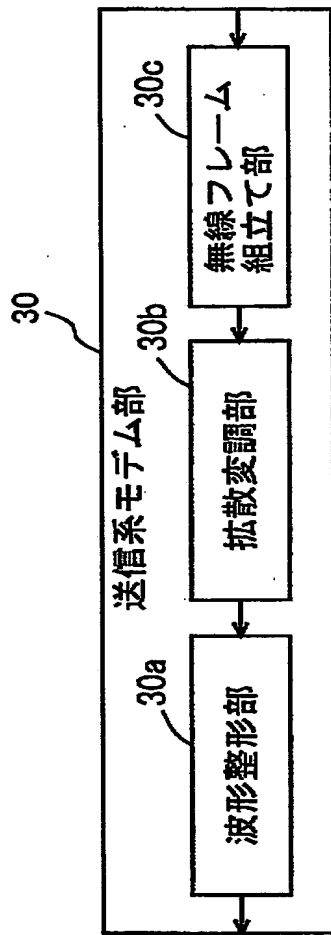
【書類名】

図面

【図1】



【図 2】



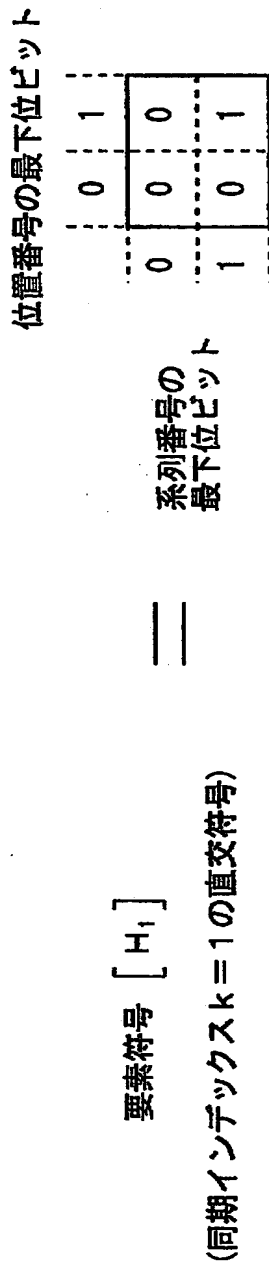
【図 3】

周期インデックス :0	$[H_0] = [0]$
周期インデックス :1	$[H_1] = \begin{bmatrix} H_0 & H_0 \\ H_0 & \overline{H_0} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
周期インデックス :2	$[H_2] = \begin{bmatrix} H_1 & H_1 \\ H_1 & \overline{H_1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$
⋮	⋮
周期インデックス :k	$[H_k] = \begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & \overline{H_{k-1}} \end{bmatrix}$

【図 4】

		位置番号 (C)															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
系列番号 (B)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	3	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
	4	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	5	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
	6	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
	7	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
	8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	9	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
	10	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0
	11	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
	12	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
	13	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1
	14	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1
	15	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

【図 5】



【図 6】

位置番号の最下位ビットを除いた値

	000	001	010	011	100	101	110	111
000	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>
001	H <sub>1</sub>	$\overline{H_1}$	H <sub>1</sub>	$\overline{H_1}$	H <sub>1</sub>	$\overline{H_1}$	H <sub>1</sub>	$\overline{H_1}$
010	H <sub>1</sub>	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	H <sub>1</sub>	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$
011	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	H <sub>1</sub>	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	H <sub>1</sub>
100	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	$\overline{H_1}$	$\overline{H_1}$
101	H <sub>1</sub>	$\overline{H_1}$	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	H <sub>1</sub>	$\overline{H_1}$	H <sub>1</sub>
110	H <sub>1</sub>	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	$\overline{H_1}$	$\overline{H_1}$	H <sub>1</sub>	H <sub>1</sub>
111	H <sub>1</sub>	$\overline{H_1}$	$\overline{H_1}$	H <sub>1</sub>	$\overline{H_1}$	H <sub>1</sub>	H <sub>1</sub>	$\overline{H_1}$

系列番号の  
最下位ビット  
を除いた値

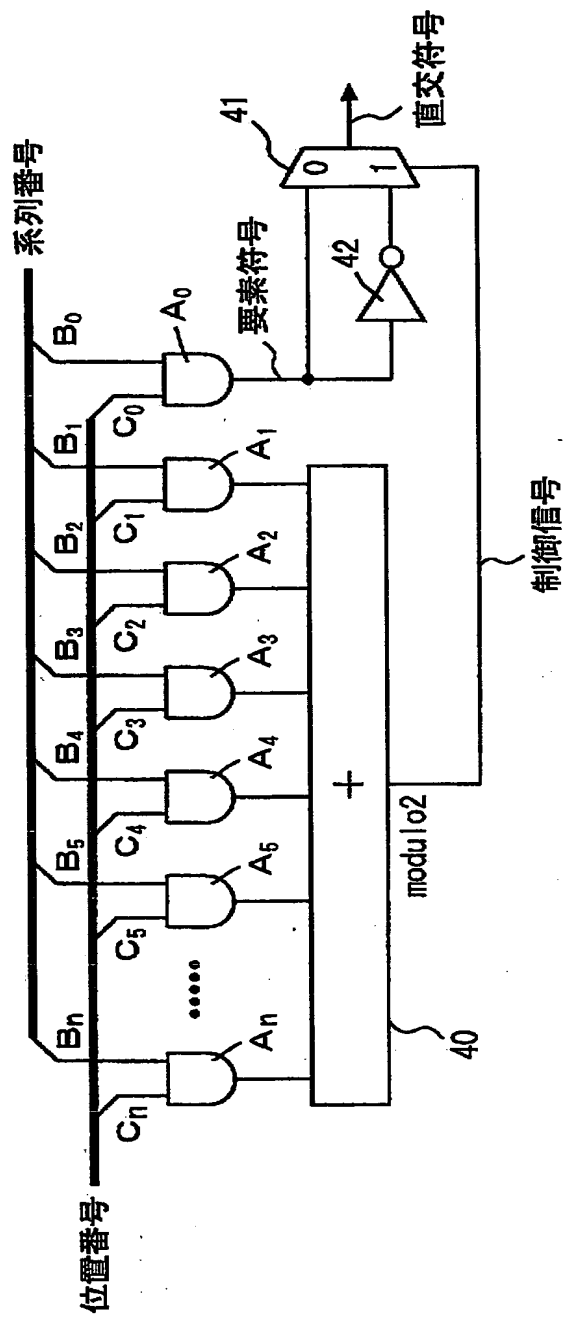
【図 7】

位置番号の最下位ビットを除いた値

	000	001	010	011	100	101	110	111
000	0	0	0	0	0	0	0	0
001	0	1	0	1	0	1	0	1
010	0	0	1	1	0	0	1	1
011	0	1	1	2	0	1	1	2
100	0	0	0	0	1	1	1	1
101	0	1	0	1	1	2	1	2
110	0	0	1	1	1	1	2	2
111	0	1	1	2	1	2	2	3

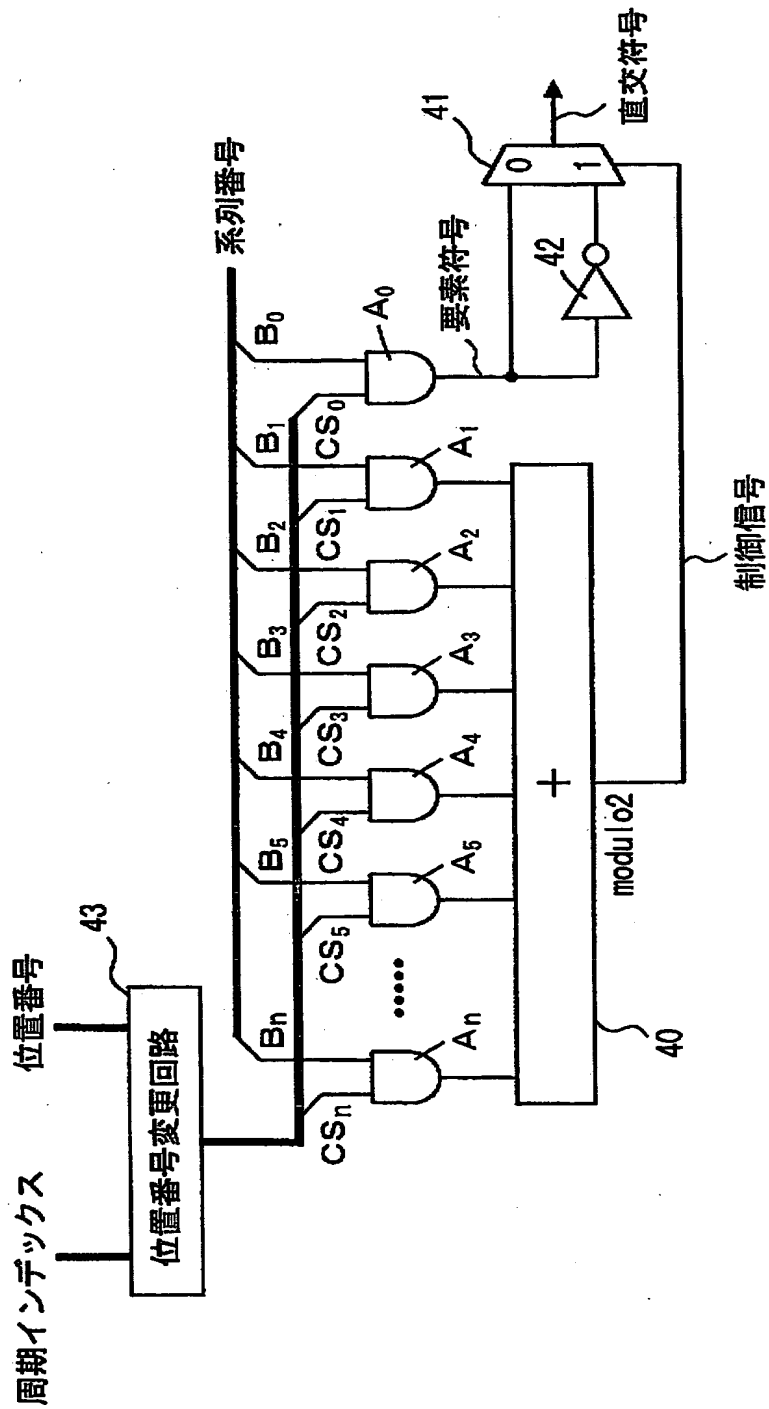
系列番号の  
最下位ビット  
を除いた値

【図 8】





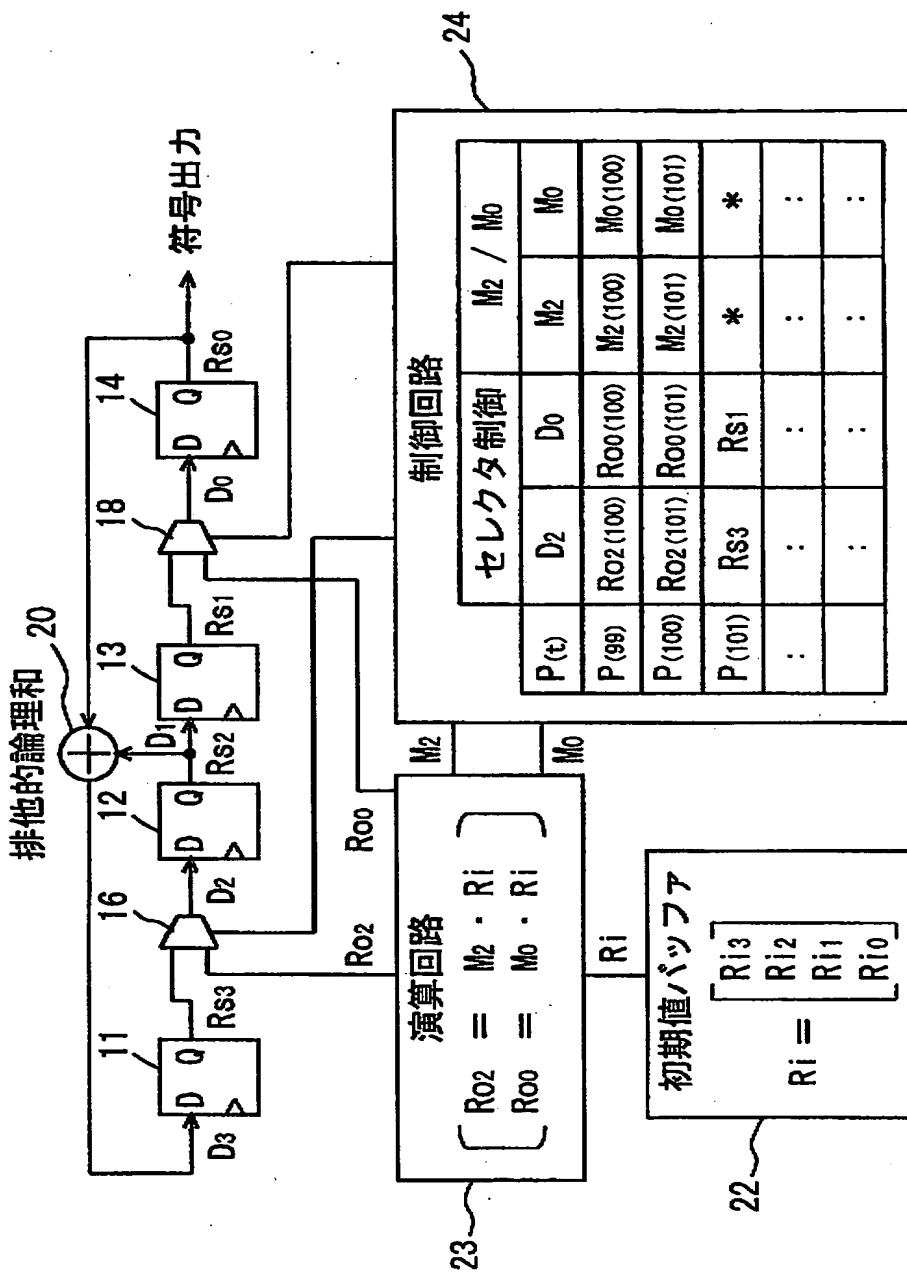
【図9】



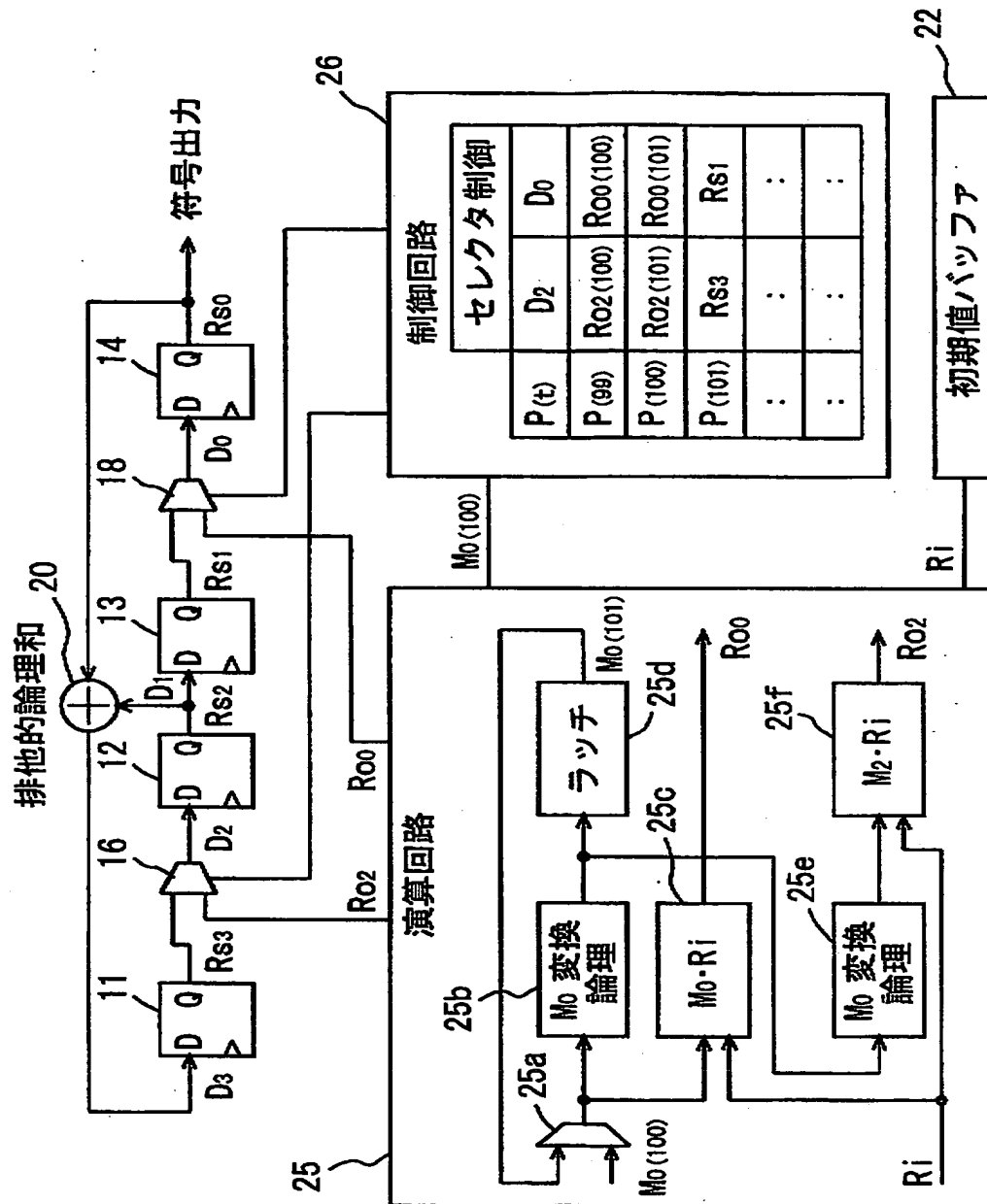
【図 1 0】

		位置番号(C)															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
系列番号 (B)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	3	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
	4	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	5	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0
	6	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
	7	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1
	8	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	9	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
	10	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
	11	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1
	12	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
	13	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
	14	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
	15	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

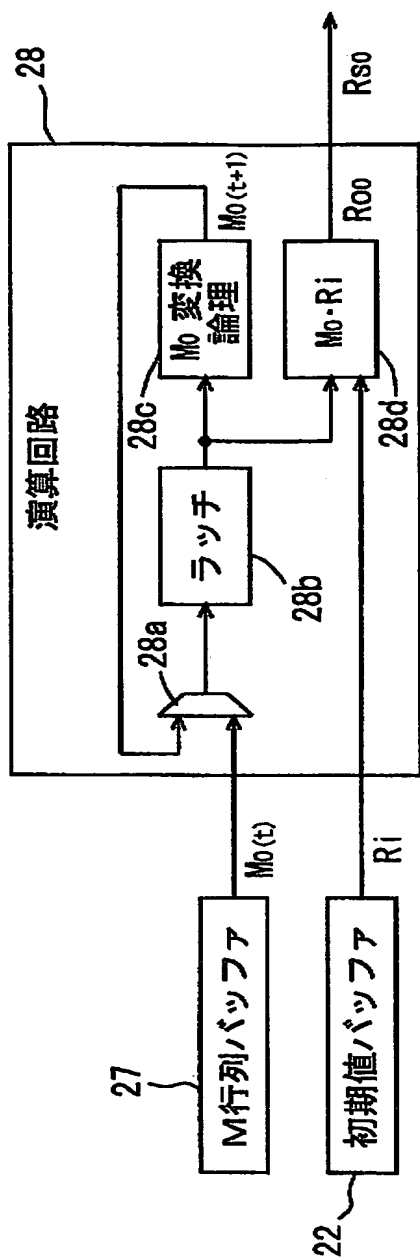
【図 11】



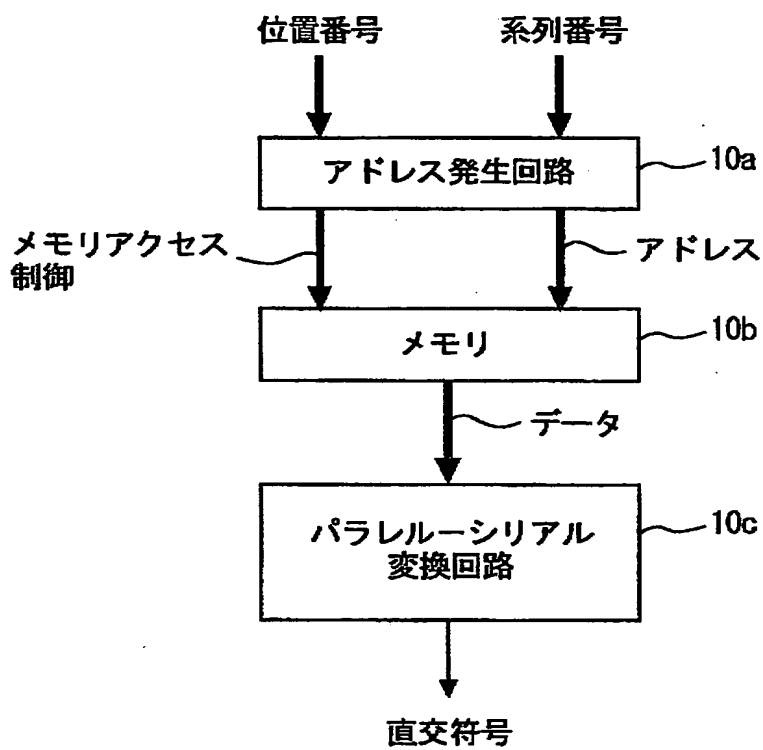
【図 12】



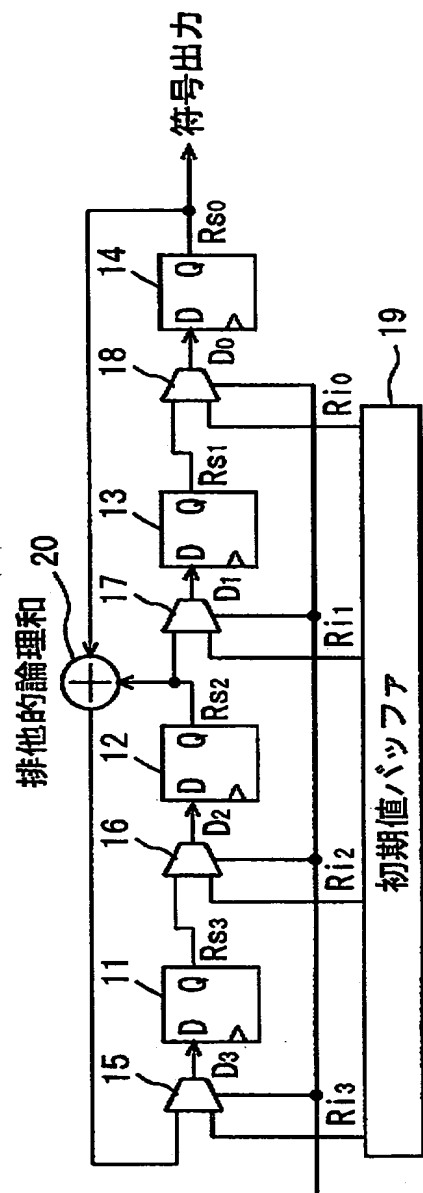
【図 1 3】



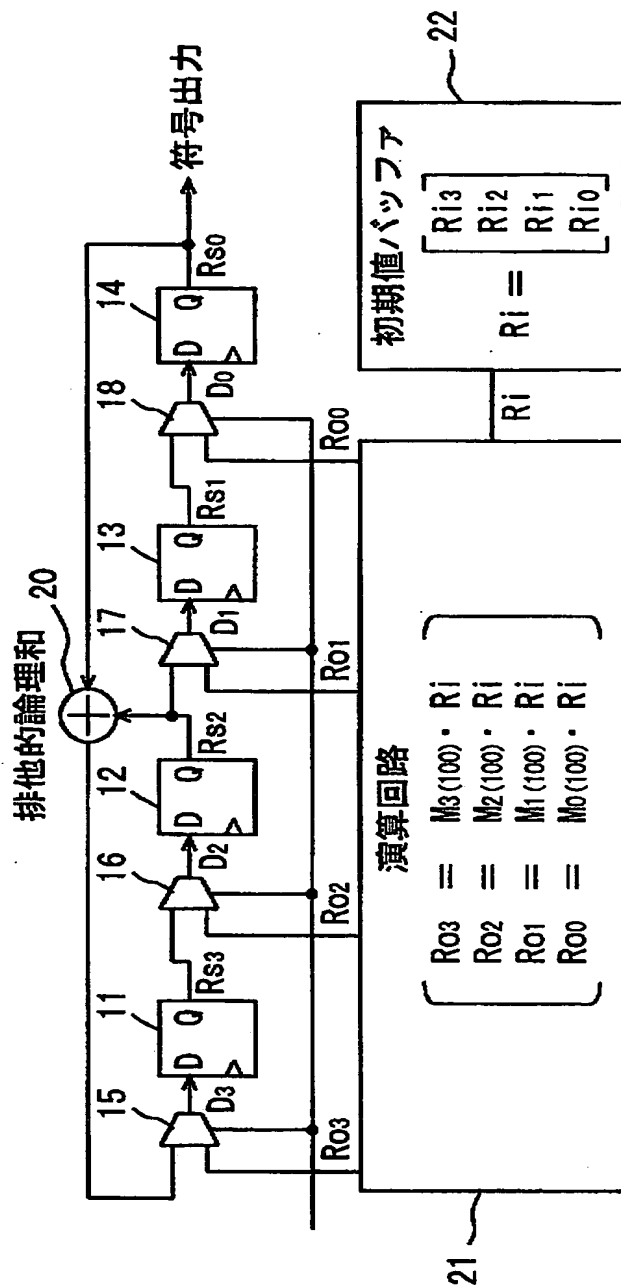
【図 1 4】



【図 15】



【図16】





【書類名】 要約書

【要約】

【課題】 回路規模の縮小を図った直交符号生成装置、スクランブル符号生成装置、およびそれらを用いた携帯無線端末を提供する。

【解決手段】 直交符号生成装置は、系列番号と位置番号とを論理積回路  $A_0$ ,  $A_1$ ,  $A_2$ ,  $A_3$ ,  $\dots$ ,  $A_n$  および加算回路 40 で論理演算することにより、メモリに予め格納することなく、直交符号系列を発生させることができる。一方、スクランブル符号生成装置は、帰還処理および拡散処理に関与するレジスタ 12, 14 の値のみを演算し、シフトレジスタをシフトさせながらレジスタ 12, 14 にロードする。レジスタ 11 ~ 14 がすべて有効な値で満たされた後は、シフトレジスタのシフト動作により、スクランブル符号を生成する。

【選択図】 図 8

出 願 人 履 歴 情 報

識別番号 [000001889]

1. 変更年月日 1993年10月20日

[変更理由] 住所変更

住 所 大阪府守口市京阪本通2丁目5番5号  
氏 名 三洋電機株式会社